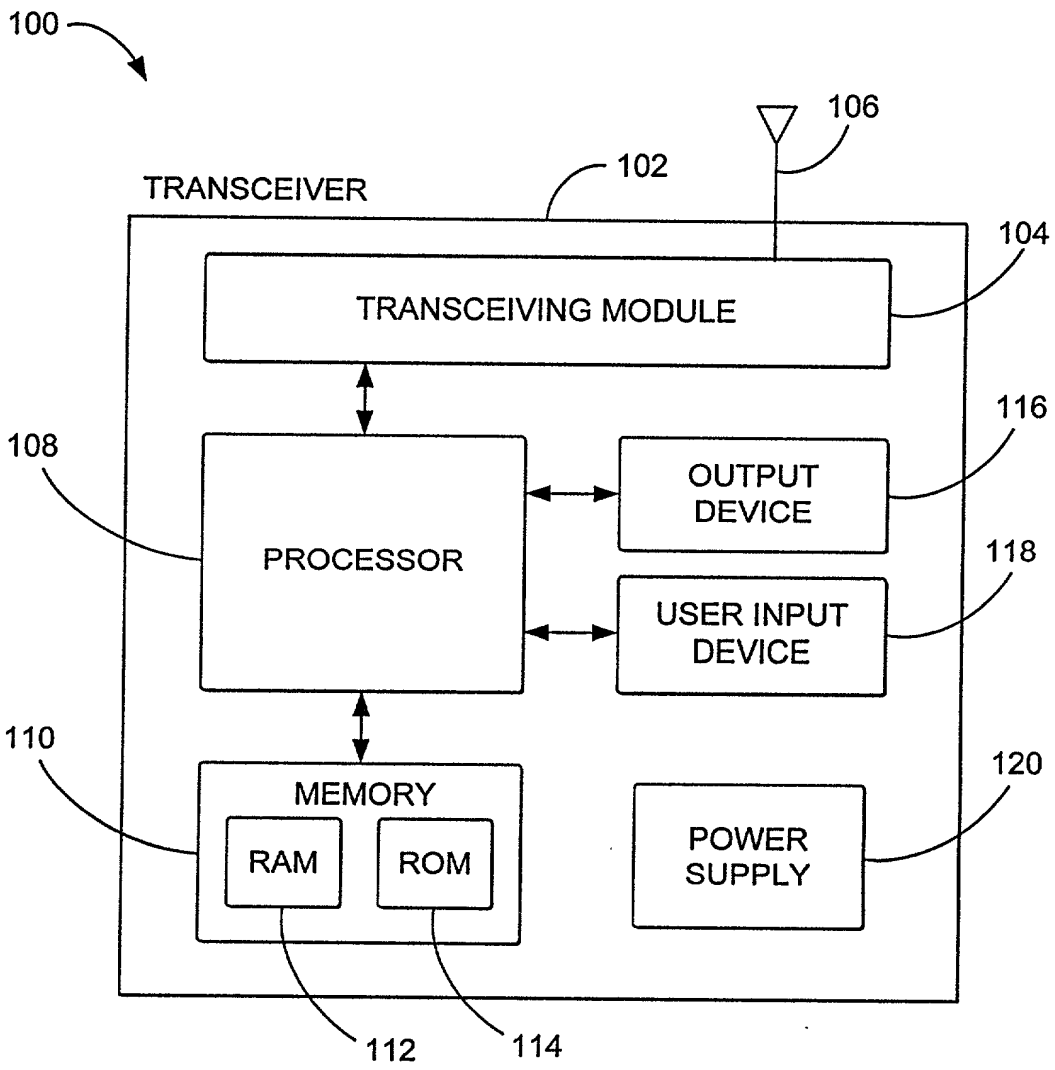

100
106
TRANSCEIVER
102
104
TRANSCEIVING MODULE
116
108
OUTPUT DEVICE
PROCESSOR
118
USER INPUT DEVICE
110
MEMORY
120
RAM
ROM
POWER SUPPLY
112
114

FIG. 1

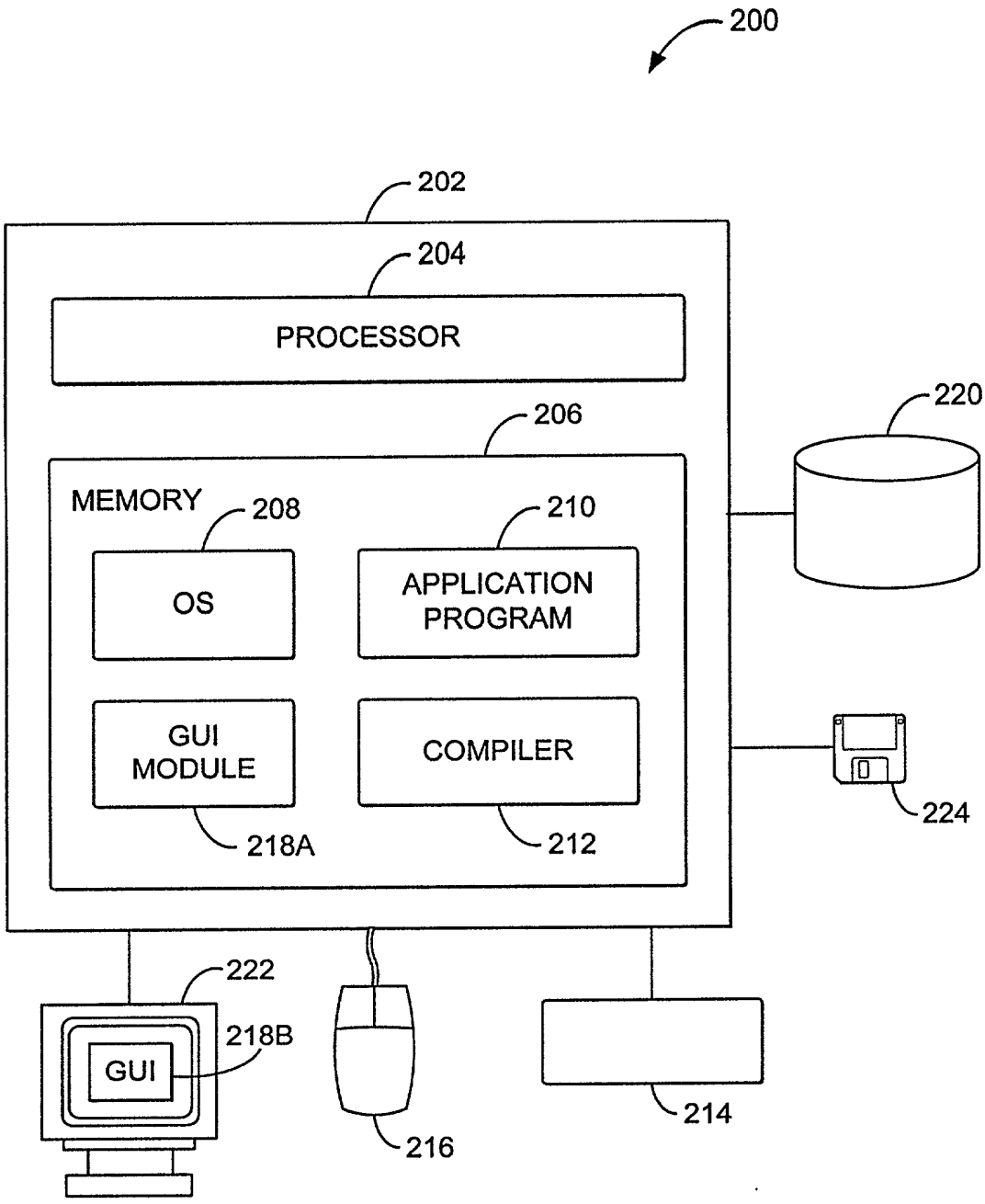

200
202
204
PROCESSOR
206
MEMORY
208
OS
210
APPLICATION PROGRAM
GUI MODULE
COMPILER
218A
212
220
224
222
218B
GUI
216
214

FIG. 2

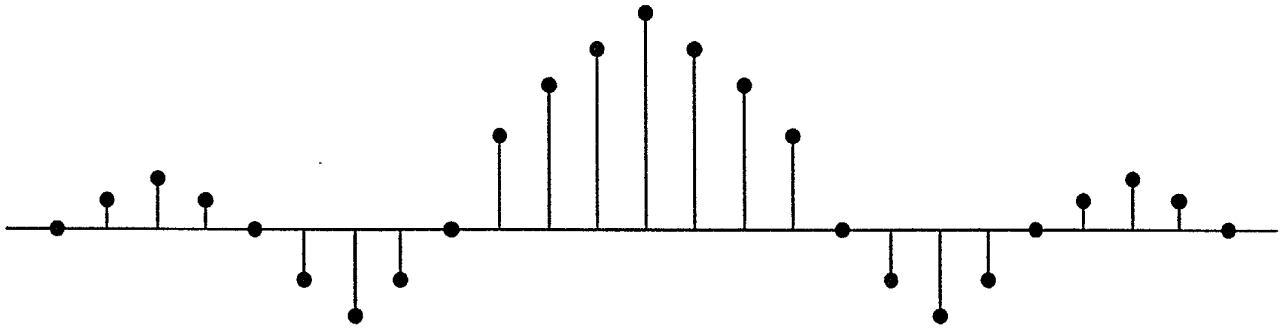


FIG. 3

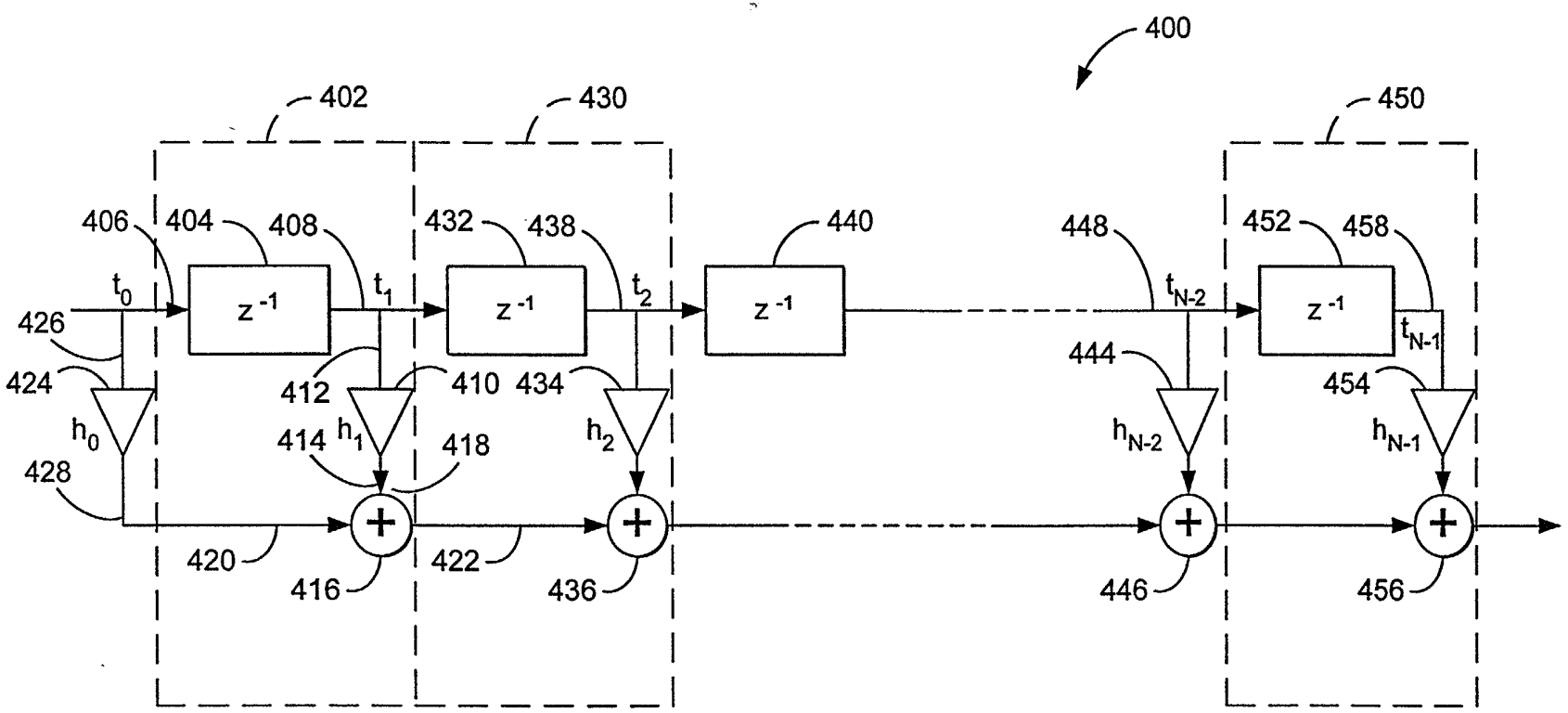


FIG. 4
Prior Art

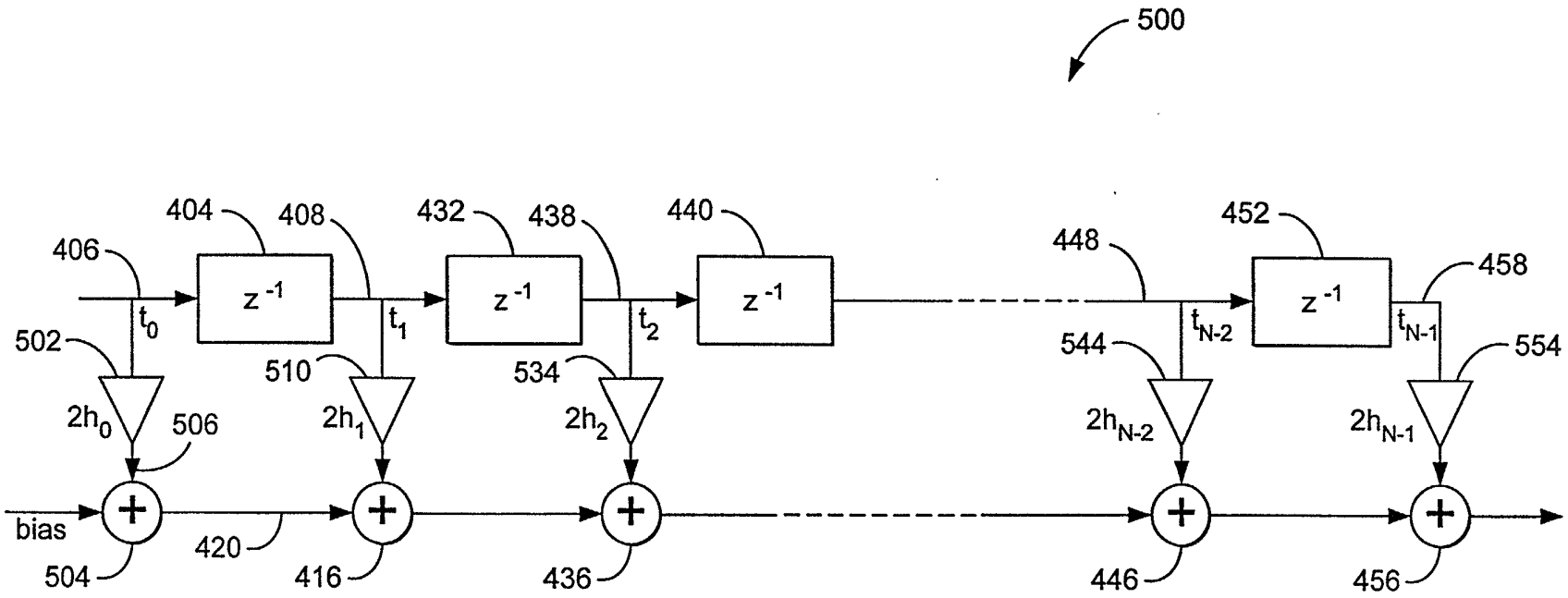


FIG. 5A

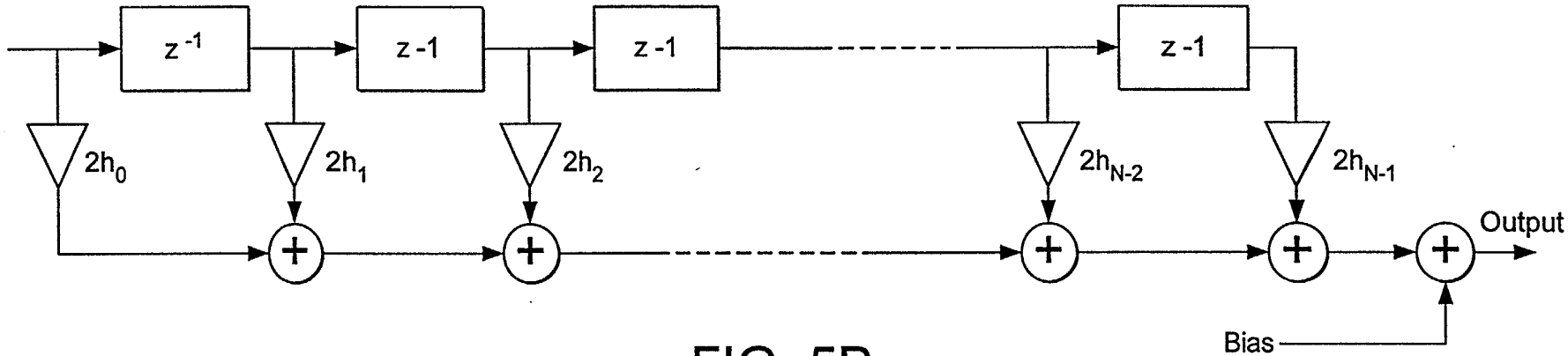

z -1
z -1
z -1
z -1
2h0
2h1
2h2
2hN-2
2hN-1
Output
Bias

FIG. 5B

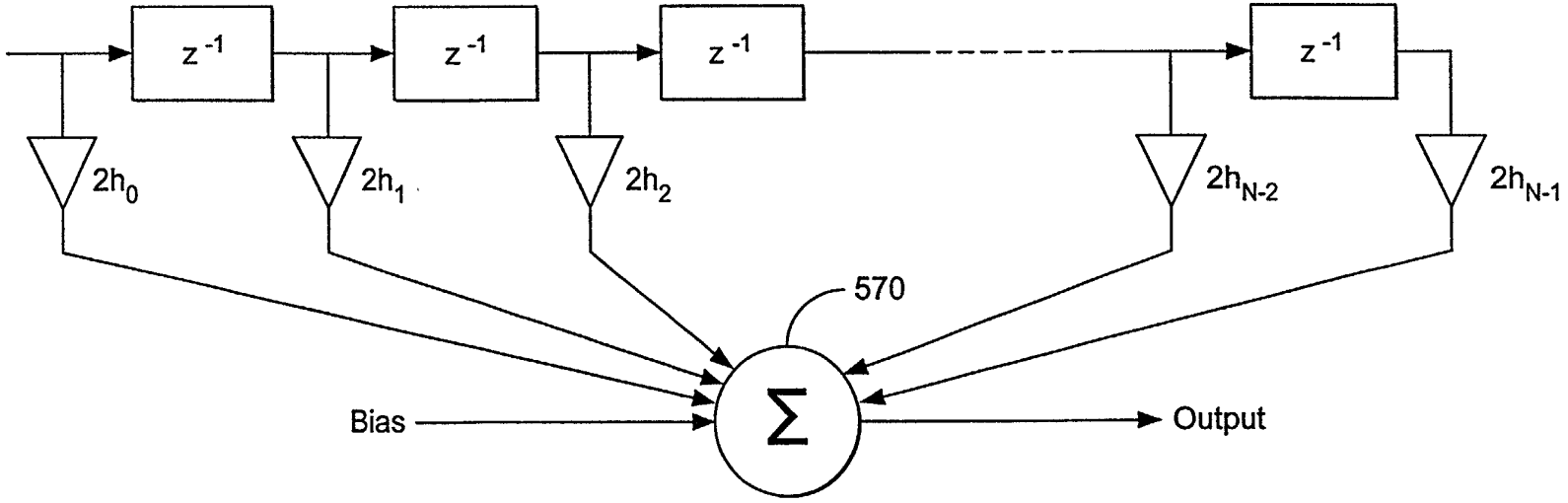

z -1
z -1
z -1
z -1
2h0
2h1
2h2
2hN-2
2hN-1
570
Σ
Bias
Output

FIG. 5C

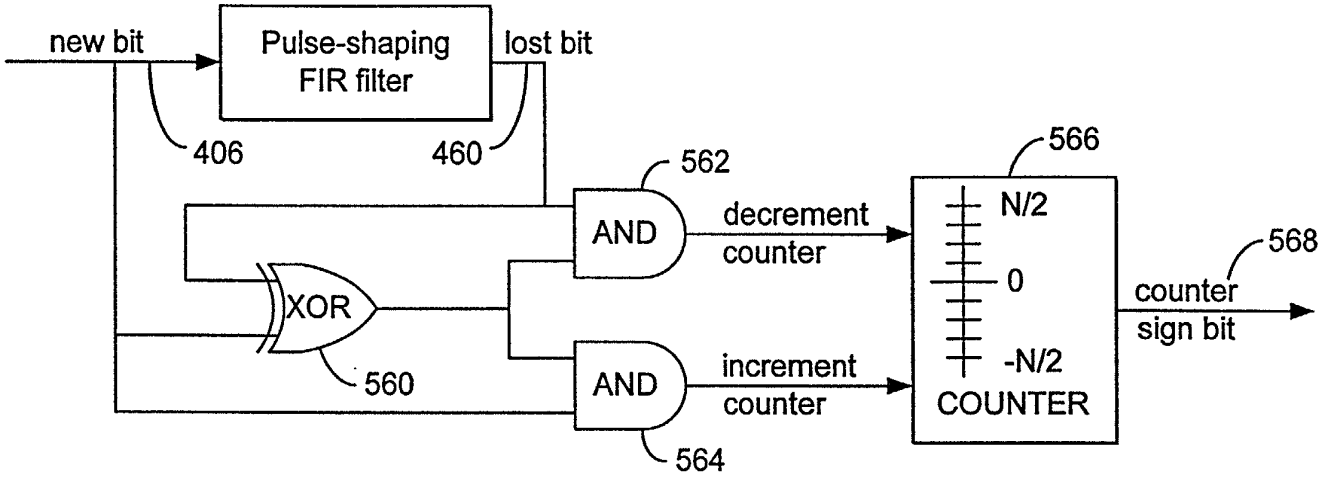

new bit
Pulse-shaping FIR filter
lost bit
406
460
562
566
AND
decrement counter
N/2
0
-N/2
COUNTER
568
counter sign bit
XOR
560
AND
increment counter
564

FIG. 5D

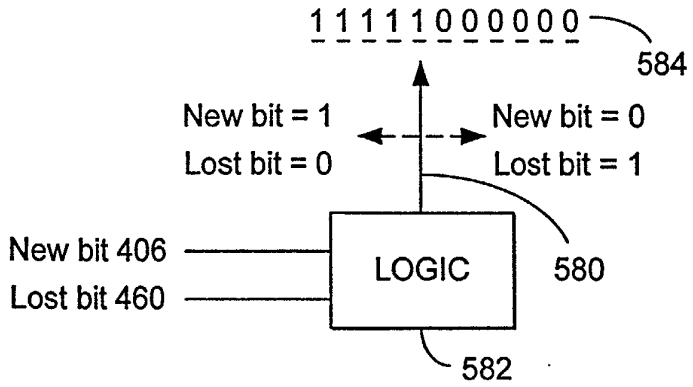

1 1 1 1 1 0 0 0 0 0 0
584
New bit = 1
Lost bit = 0
New bit = 0
Lost bit = 1
New bit 406
Lost bit 460
LOGIC
580
582

FIG. 5E

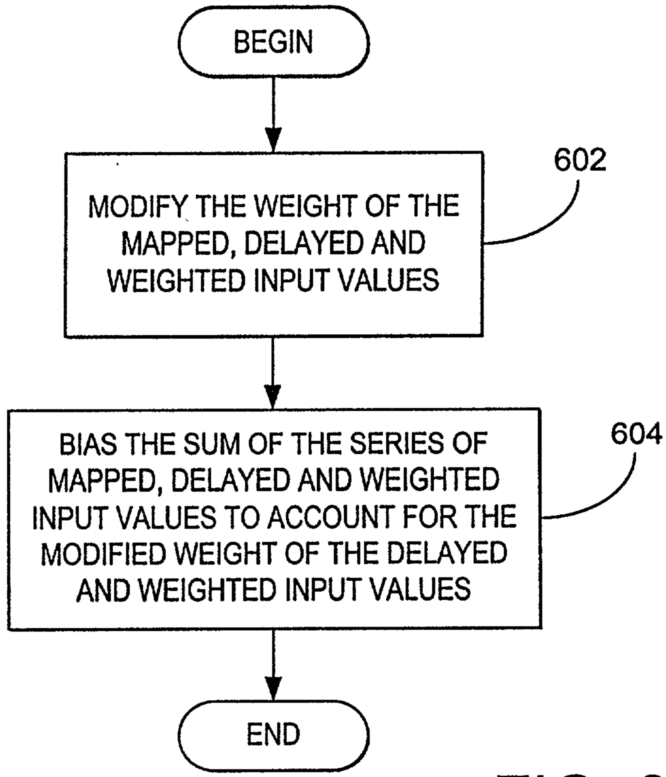

BEGIN
MODIFY THE WEIGHT OF THE MAPPED, DELAYED AND WEIGHTED INPUT VALUES
602
BIAS THE SUM OF THE SERIES OF MAPPED, DELAYED AND WEIGHTED INPUT VALUES TO ACCOUNT FOR THE MODIFIED WEIGHT OF THE DELAYED AND WEIGHTED INPUT VALUES
604
END

FIG. 6A

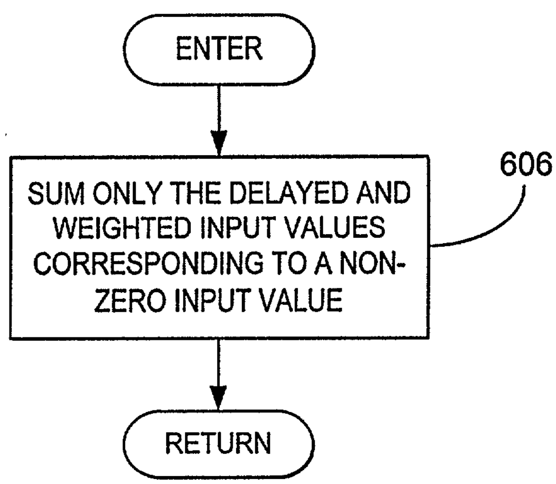

ENTER
SUM ONLY THE DELAYED AND WEIGHTED INPUT VALUES CORRESPONDING TO A NON-ZERO INPUT VALUE
606
RETURN

FIG. 6B

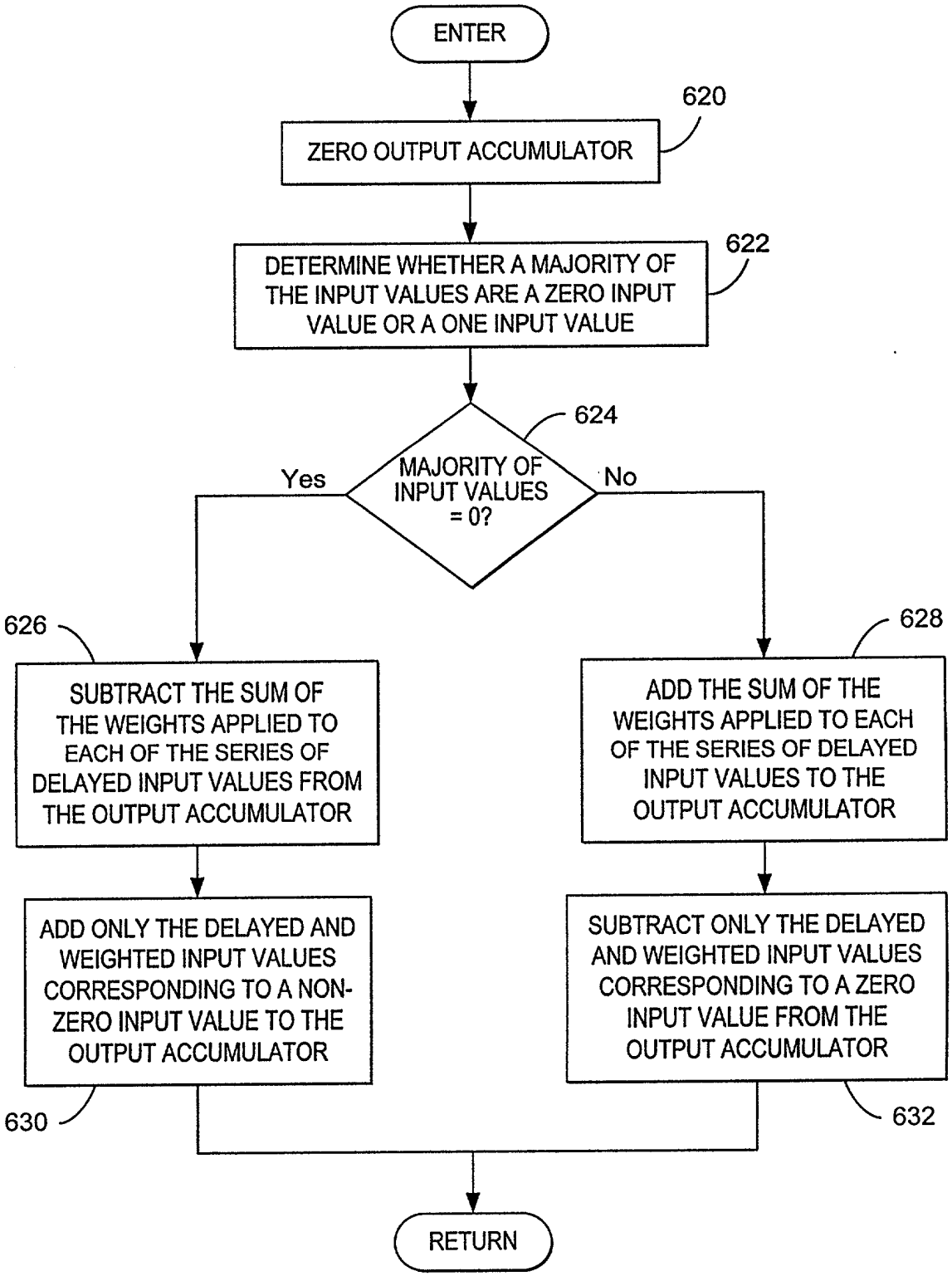

ENTER
620
ZERO OUTPUT ACCUMULATOR
622
DETERMINE WHETHER A MAJORITY OF THE INPUT VALUES ARE A ZERO INPUT VALUE OR A ONE INPUT VALUE
624
MAJORITY OF INPUT VALUES = 0?
Yes
No
626
SUBTRACT THE SUM OF THE WEIGHTS APPLIED TO EACH OF THE SERIES OF DELAYED INPUT VALUES FROM THE OUTPUT ACCUMULATOR
628
ADD THE SUM OF THE WEIGHTS APPLIED TO EACH OF THE SERIES OF DELAYED INPUT VALUES TO THE OUTPUT ACCUMULATOR
ADD ONLY THE DELAYED AND WEIGHTED INPUT VALUES CORRESPONDING TO A NON-ZERO INPUT VALUE TO THE OUTPUT ACCUMULATOR
SUBTRACT ONLY THE DELAYED AND WEIGHTED INPUT VALUES CORRESPONDING TO A ZERO INPUT VALUE FROM THE OUTPUT ACCUMULATOR
630
632
RETURN

FIG. 6C

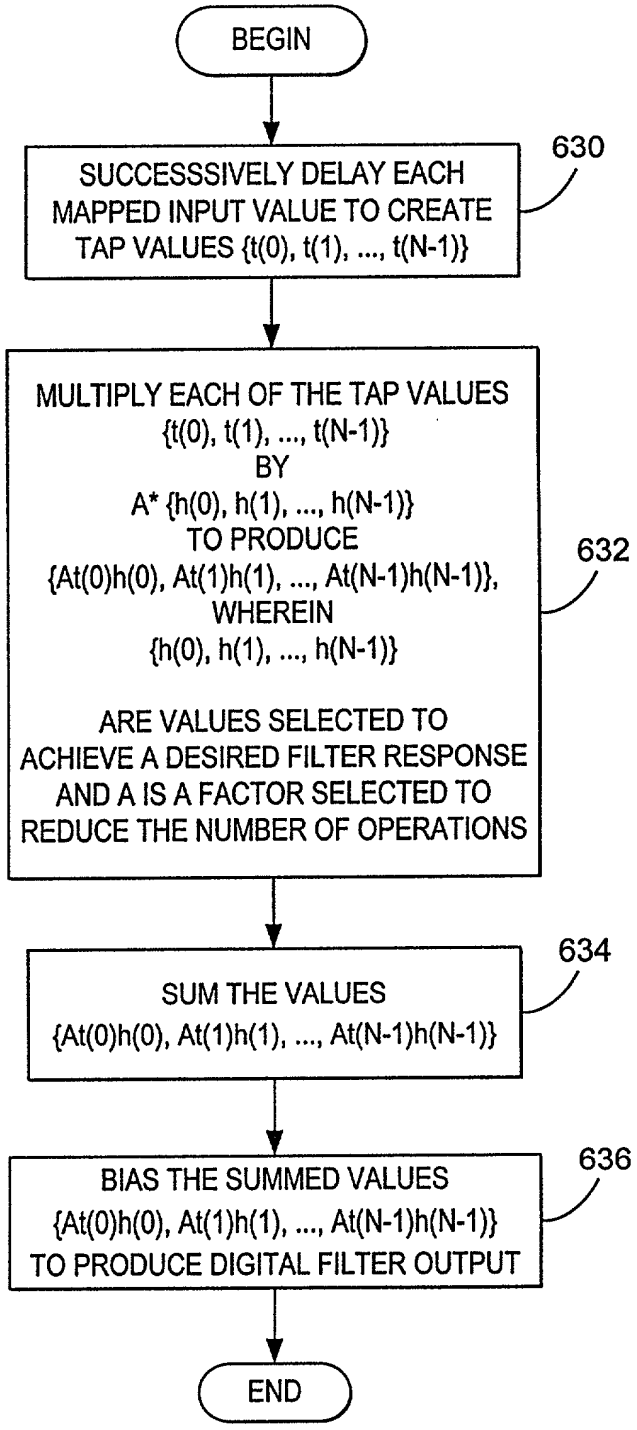

BEGIN
SUCCESSSIVELY DELAY EACH MAPPED INPUT VALUE TO CREATE TAP VALUES {t(0), t(1), ..., t(N-1)}
630
MULTIPLY EACH OF THE TAP VALUES {t(0), t(1), ..., t(N-1)} BY A* {h(0), h(1), ..., h(N-1)} TO PRODUCE {At(0)h(0), At(1)h(1), ..., At(N-1)h(N-1)}, WHEREIN {h(0), h(1), ..., h(N-1)} ARE VALUES SELECTED TO ACHIEVE A DESIRED FILTER RESPONSE AND A IS A FACTOR SELECTED TO REDUCE THE NUMBER OF OPERATIONS
632
SUM THE VALUES {At(0)h(0), At(1)h(1), ..., At(N-1)h(N-1)}
634
BIAS THE SUMMED VALUES {At(0)h(0), At(1)h(1), ..., At(N-1)h(N-1)} TO PRODUCE DIGITAL FILTER OUTPUT
636
END

FIG. 6D

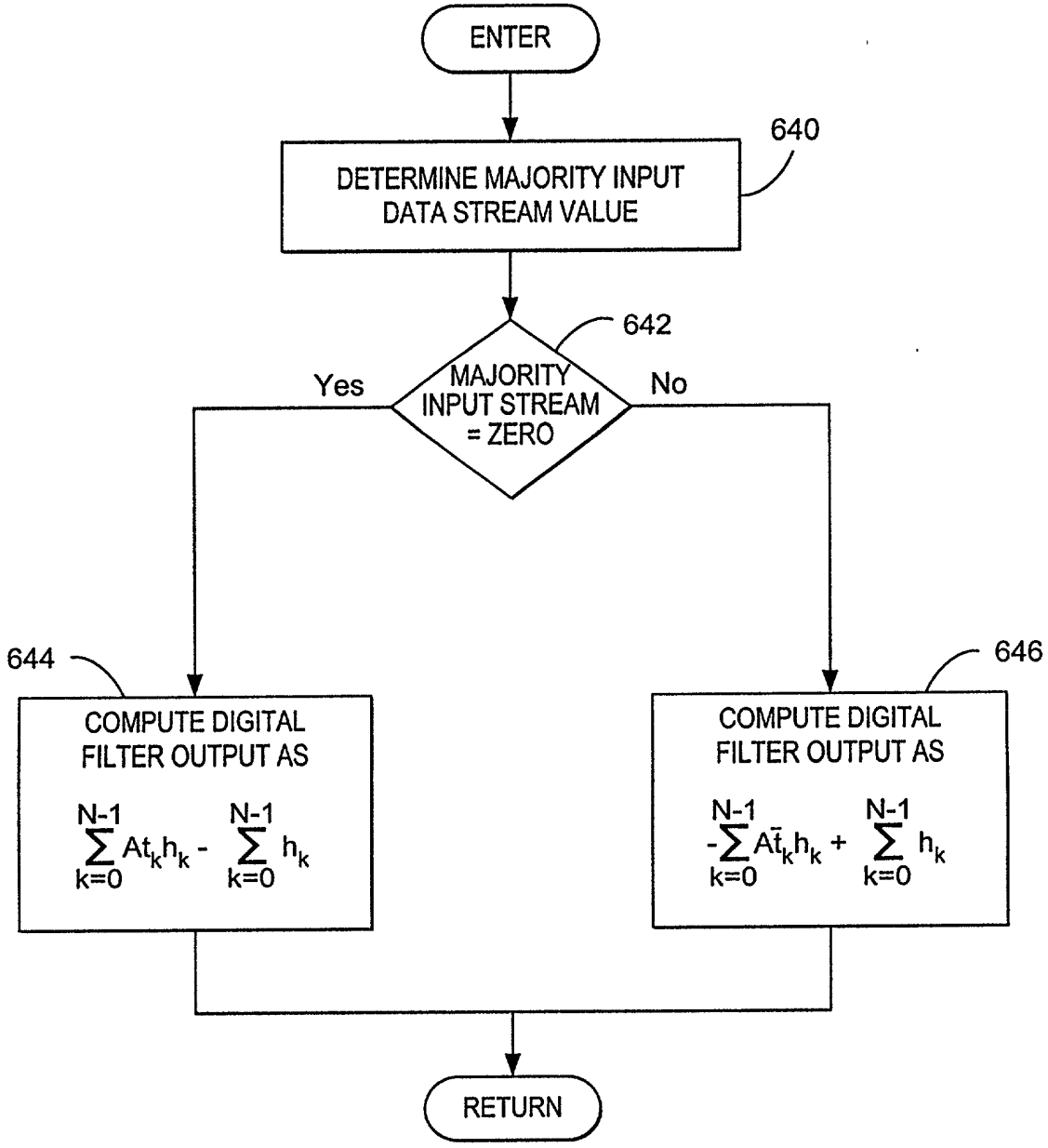

ENTER
640
DETERMINE MAJORITY INPUT DATA STREAM VALUE
642
MAJORITY INPUT STREAM = ZERO
Yes
No
644
COMPUTE DIGITAL FILTER OUTPUT AS
$\sum_{k=0}^{N-1} At_k h_k - \sum_{k=0}^{N-1} h_k$
646
COMPUTE DIGITAL FILTER OUTPUT AS
$-\sum_{k=0}^{N-1} A\bar{t}_k h_k + \sum_{k=0}^{N-1} h_k$
RETURN

FIG. 6E

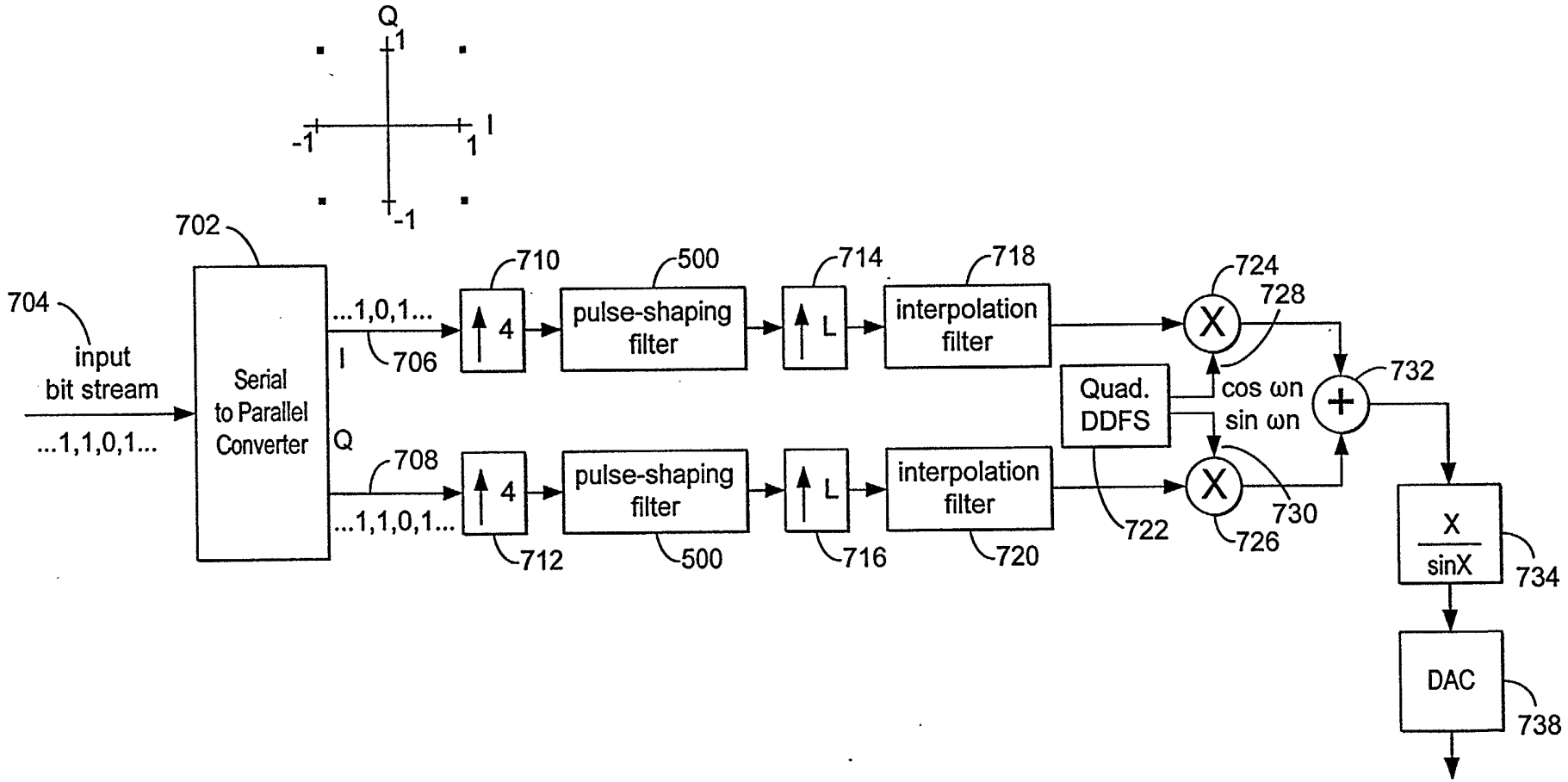
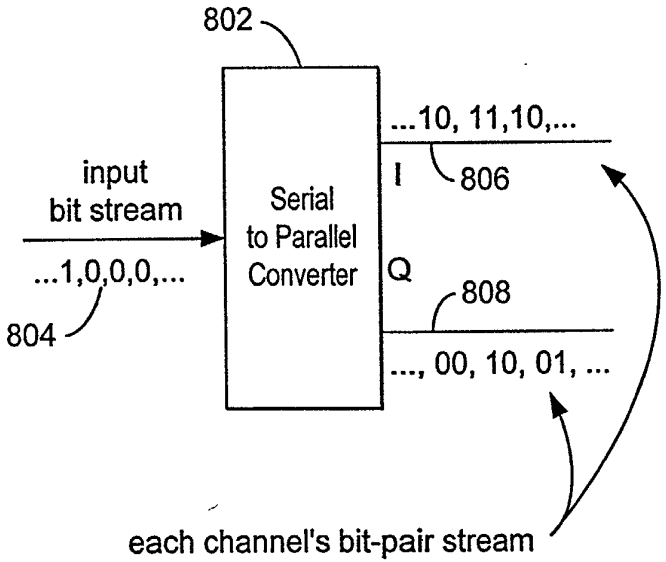

Q
1
-1
1
I
-1
702
704
input
bit stream
...1,1,0,1...
Serial
to Parallel
Converter
...1,0,1...
I
706
710
4
500
pulse-shaping
filter
714
L
718
interpolation
filter
Q
...1,1,0,1...
708
4
712
pulse-shaping
filter
500
L
716
interpolation
filter
720
722
Quad.
DDFS
cos ωn
sin ωn
724
728
730
726
732
X
sinX
734
DAC
738

FIG. 7


802
input
bit stream
...1,0,0,0,...
804
Serial
to Parallel
Converter
I
Q
...10, 11,10,...
806
808
..., 00, 10, 01, ...
each channel's bit-pair stream

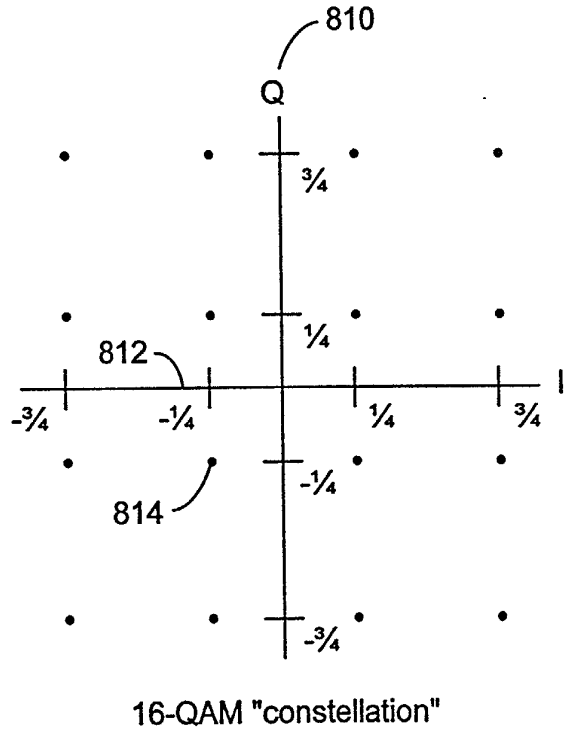

810
Q
3/4
1/4
812
I
-3/4
-1/4
1/4
3/4
-1/4
814
-3/4
16-QAM "constellation"

FIG. 8

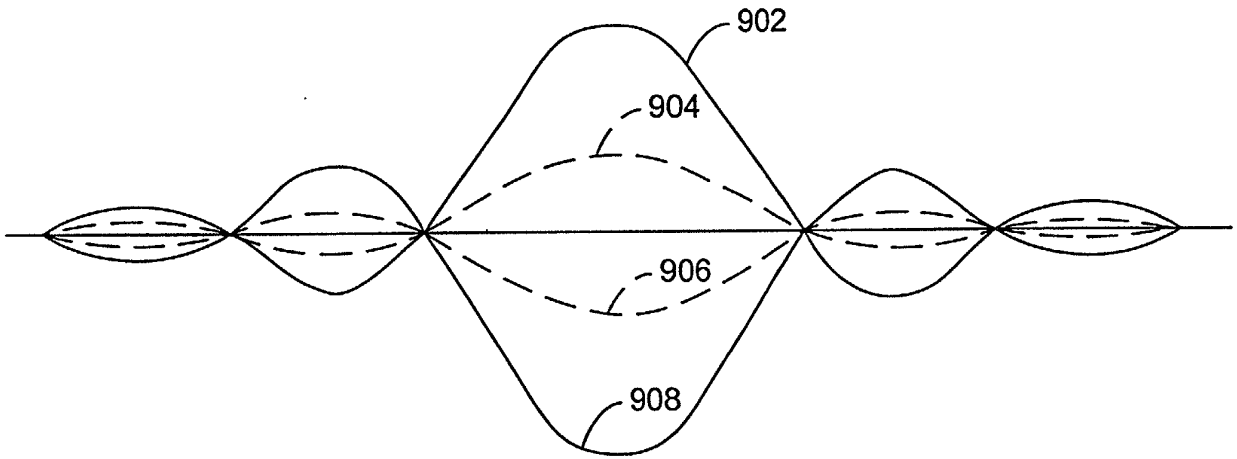


FIG. 9

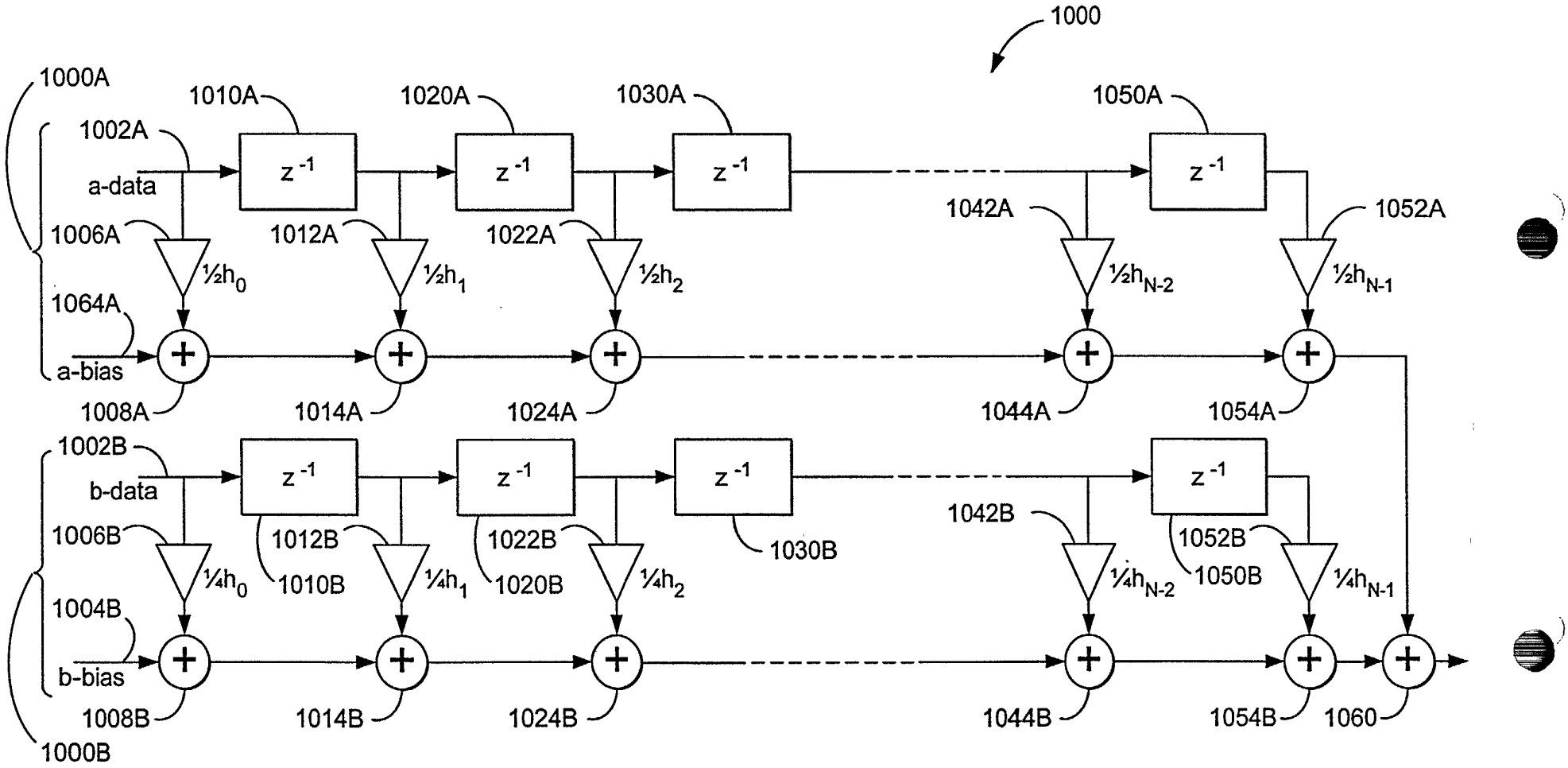


FIG. 10

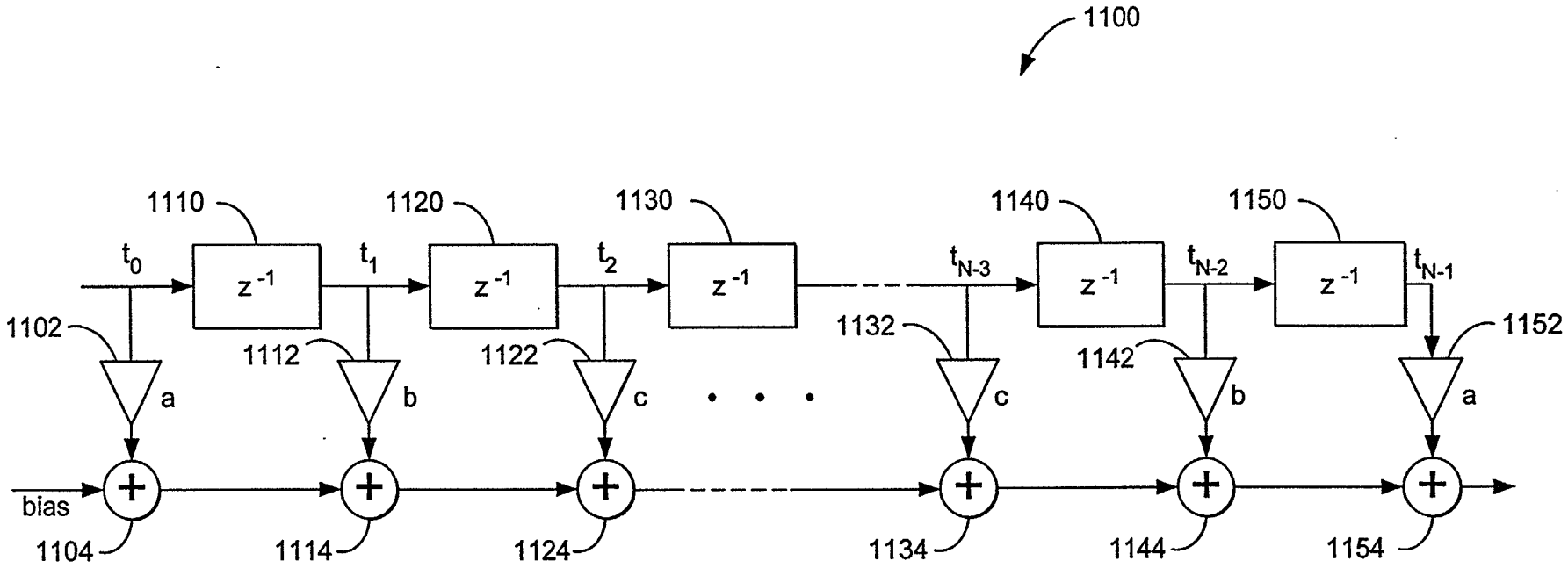


FIG. 11

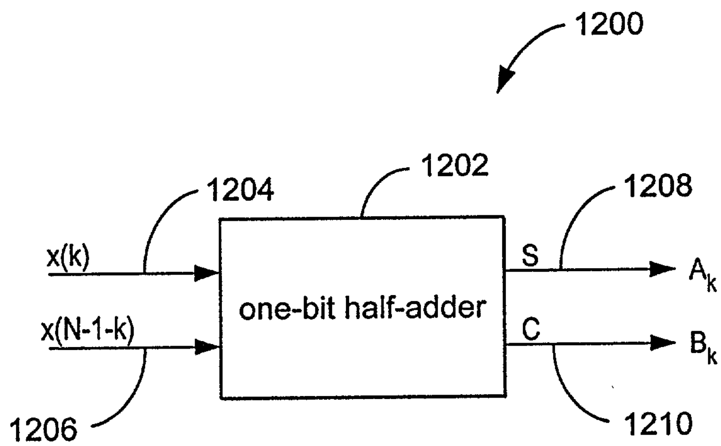

1200
1202
1204
1208
x(k)
S
A_k
one-bit half-adder
x(N-1-k)
C
B_k
1206
1210

FIG. 12

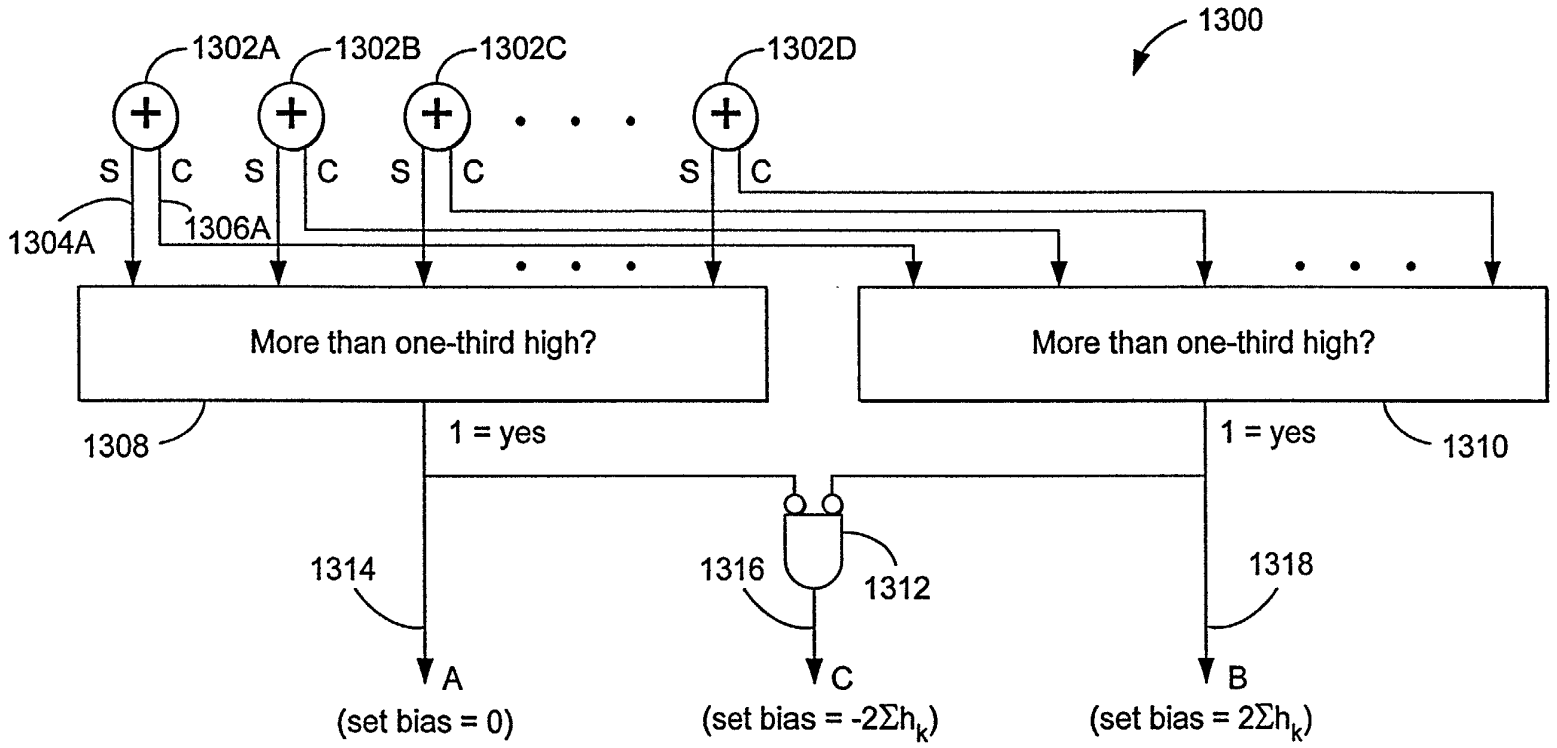

1300
1302A
1302B
1302C
1302D
S
C
1304A
1306A
More than one-third high?
More than one-third high?
1308
1310
1 = yes
1 = yes
1314
1316
1312
1318
A
C
B
(set bias = 0)
(set bias = -2Σh_k)
(set bias = 2Σh_k)

FIG. 13

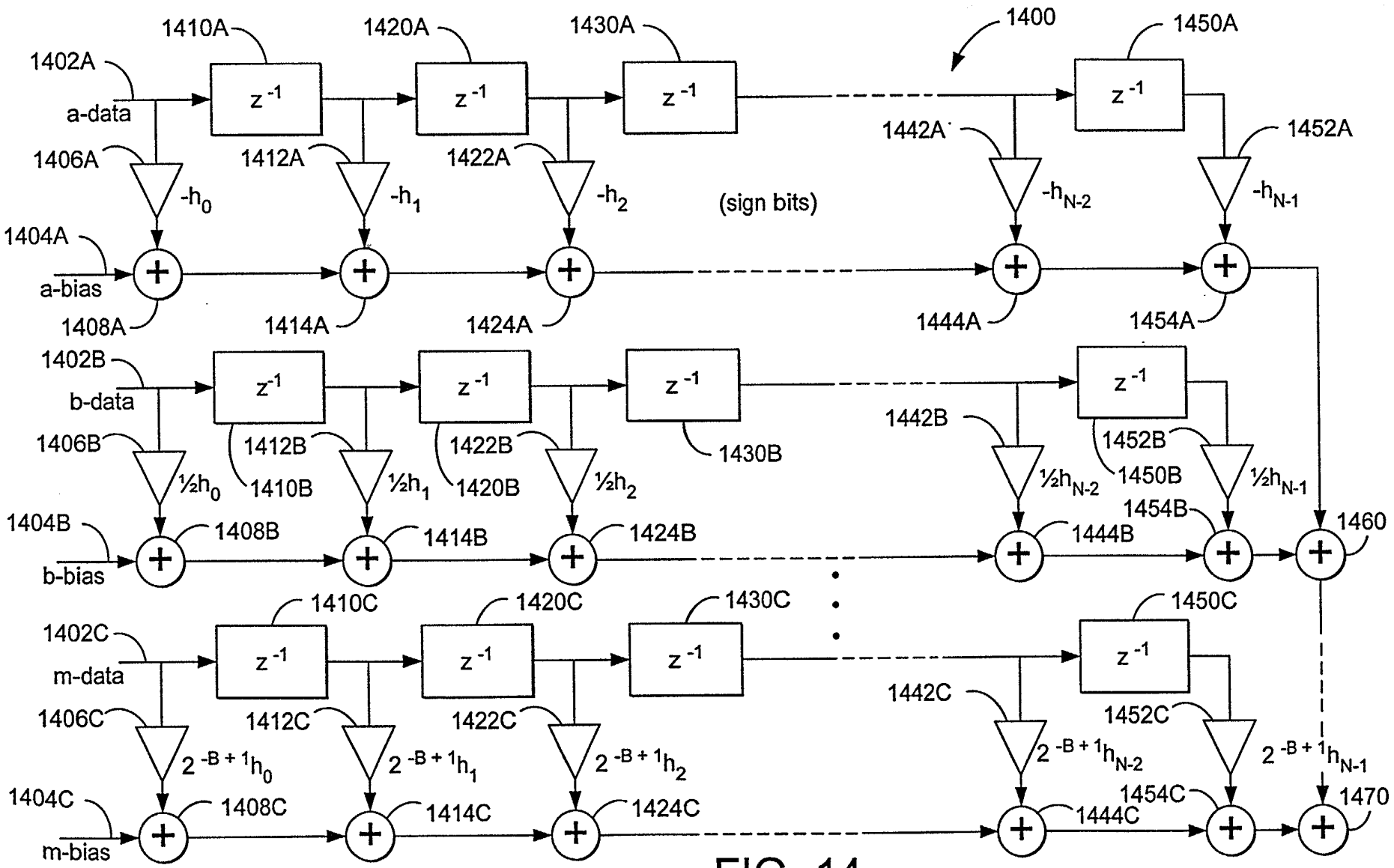

1400
1402A
a-data
1404A
a-bias
1406A
1408A
1410A
1412A
1414A
1420A
1422A
1424A
1430A
1442A
1444A
1450A
1452A
1454A
(sign bits)
1402B
b-data
1404B
b-bias
1406B
1408B
1410B
1412B
1414B
1420B
1422B
1424B
1430B
1442B
1444B
1450B
1452B
1454B
1460
1402C
m-data
1404C
m-bias
1406C
1408C
1410C
1412C
1414C
1420C
1422C
1424C
1430C
1442C
1444C
1450C
1452C
1454C
1470

FIG. 14

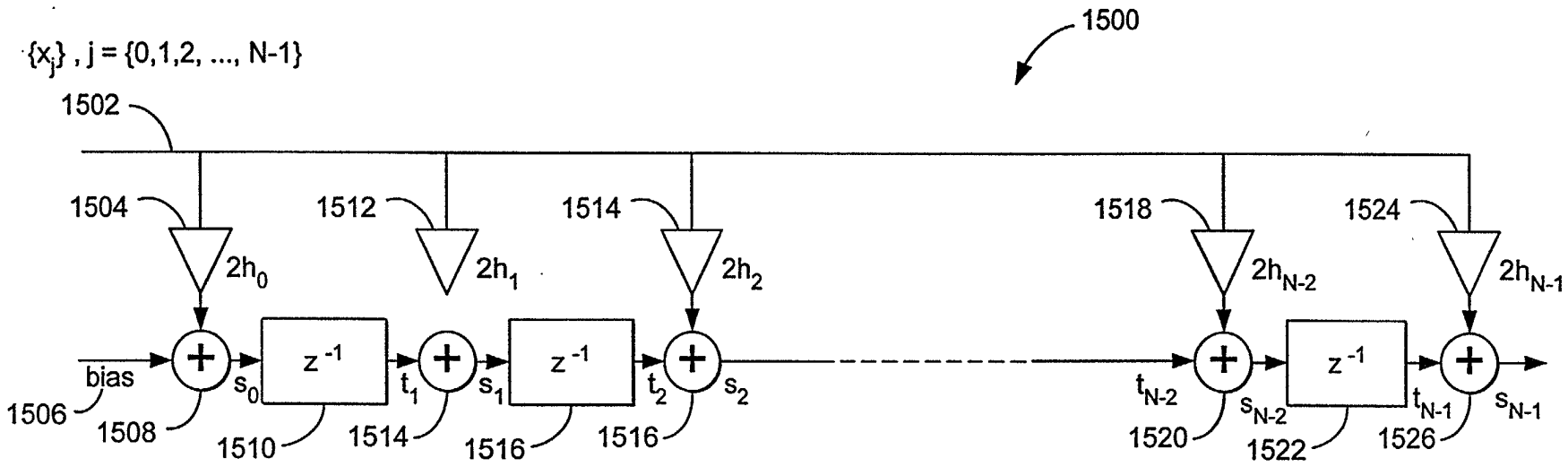


FIG. 15

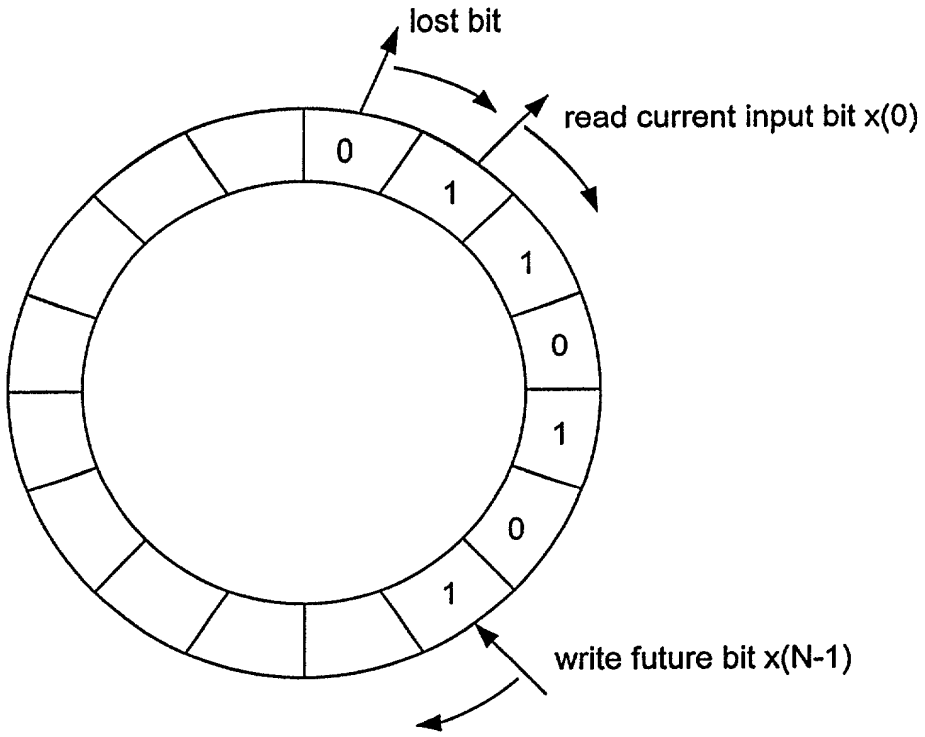

lost bit
read current input bit x(0)
0
1
1
0
1
0
1
write future bit x(N-1)

FIG. 16

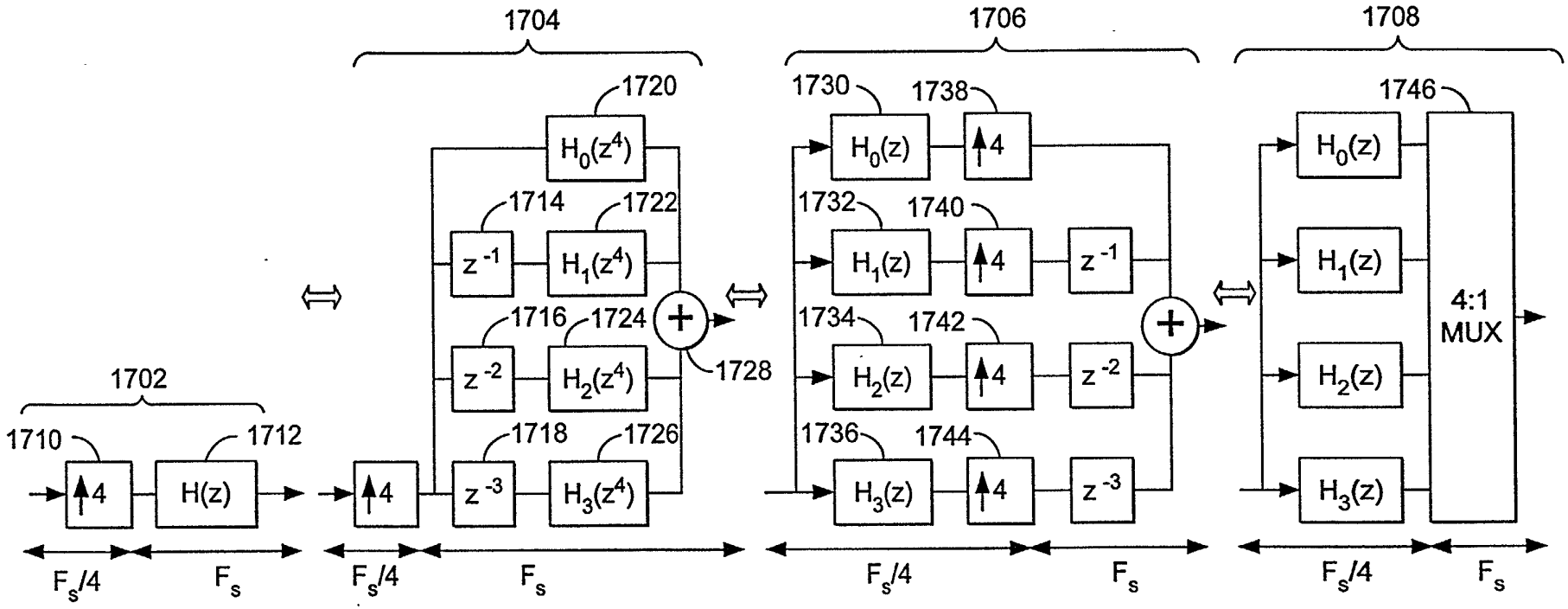

1702
1704
1706
1708
1710
1712
1714
1716
1718
1720
1722
1724
1726
1728
1730
1732
1734
1736
1738
1740
1742
1744
1746
4
H(z)
z^{-1}
z^{-2}
z^{-3}
$H_0(z^4)$
$H_1(z^4)$
$H_2(z^4)$
$H_3(z^4)$
$H_0(z)$
$H_1(z)$
$H_2(z)$
$H_3(z)$
4:1 MUX
$F_s/4$
F_s

FIG. 17

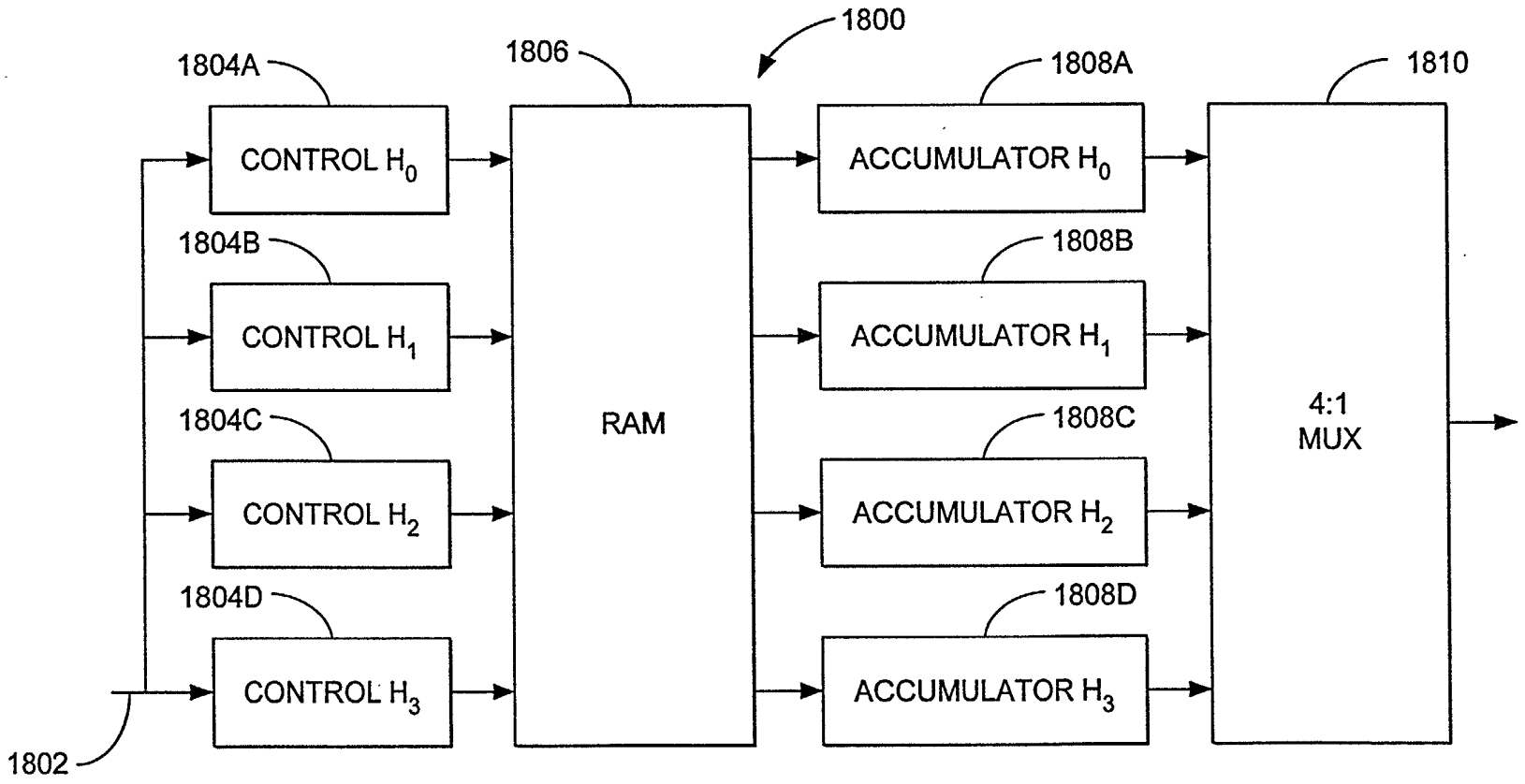

1800
1804A
1806
1808A
1810
CONTROL H_0
ACCUMULATOR H_0
1804B
1808B
CONTROL H_1
ACCUMULATOR H_1
RAM
4:1
MUX
1804C
1808C
CONTROL H_2
ACCUMULATOR H_2
1804D
1808D
CONTROL H_3
ACCUMULATOR H_3
1802

FIG. 18

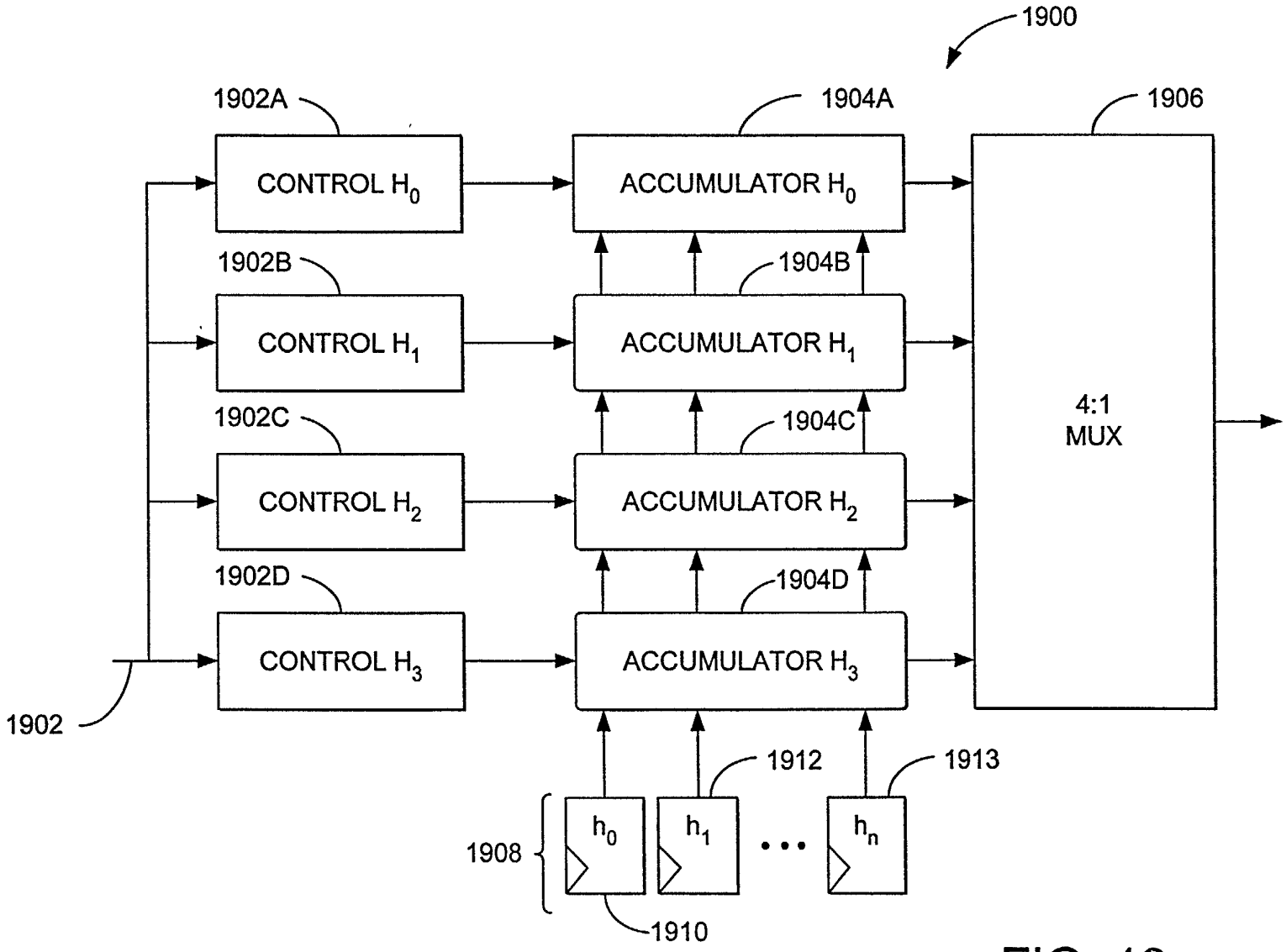
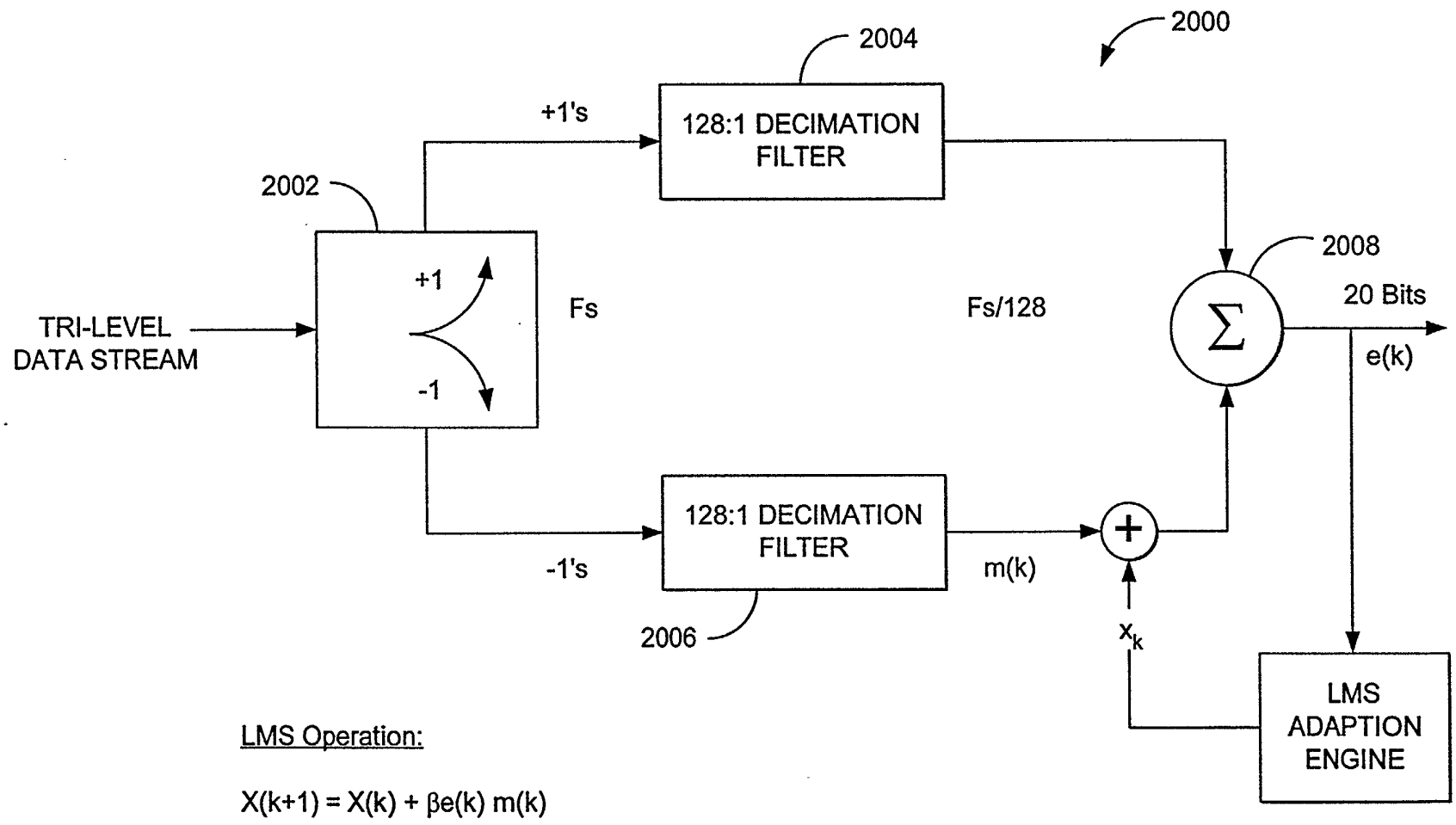


FIG. 19



LMS Operation:

$$X(k+1) = X(k) + \beta e(k)\, m(k)$$

FIG. 20

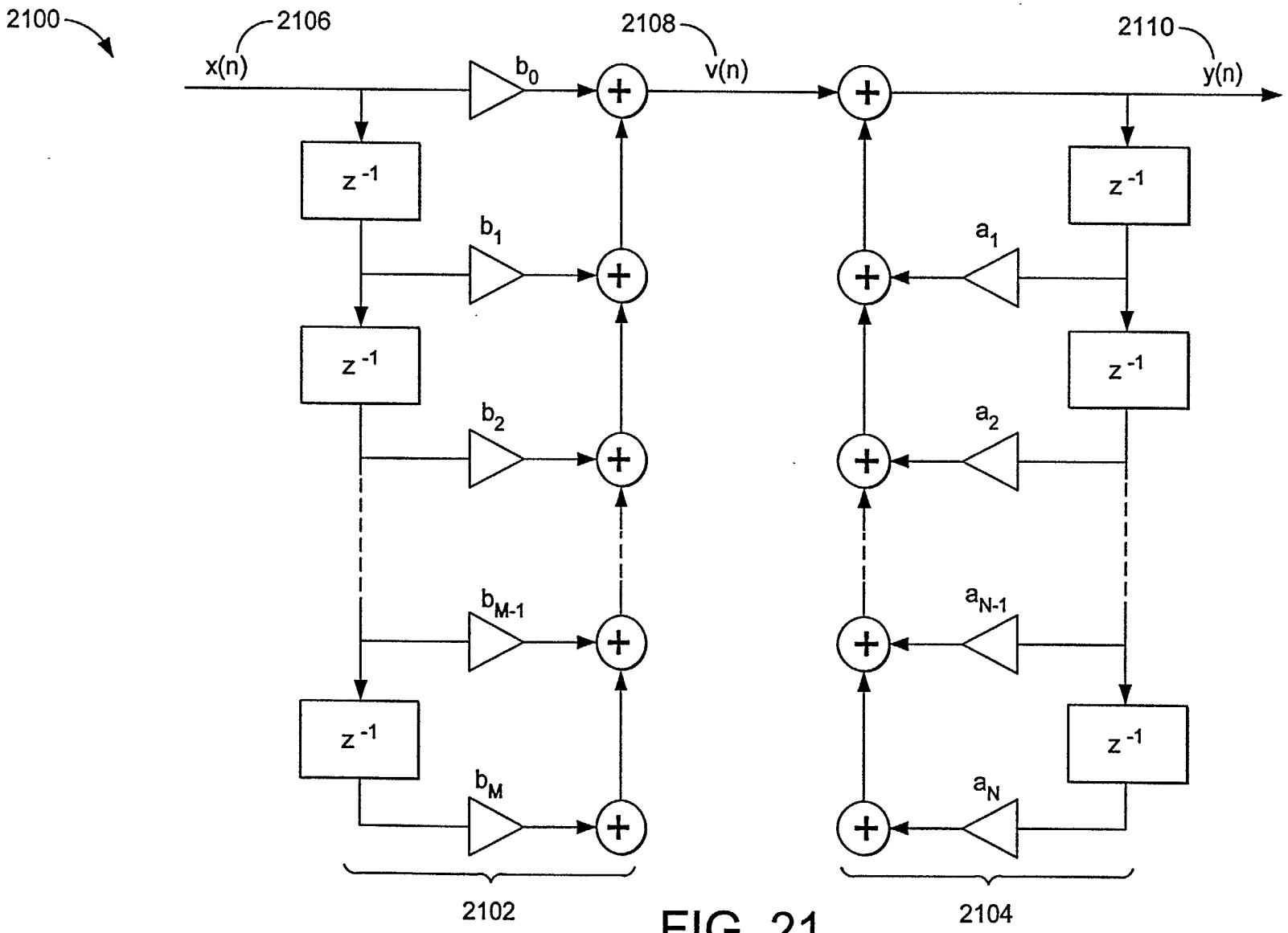


FIG. 21

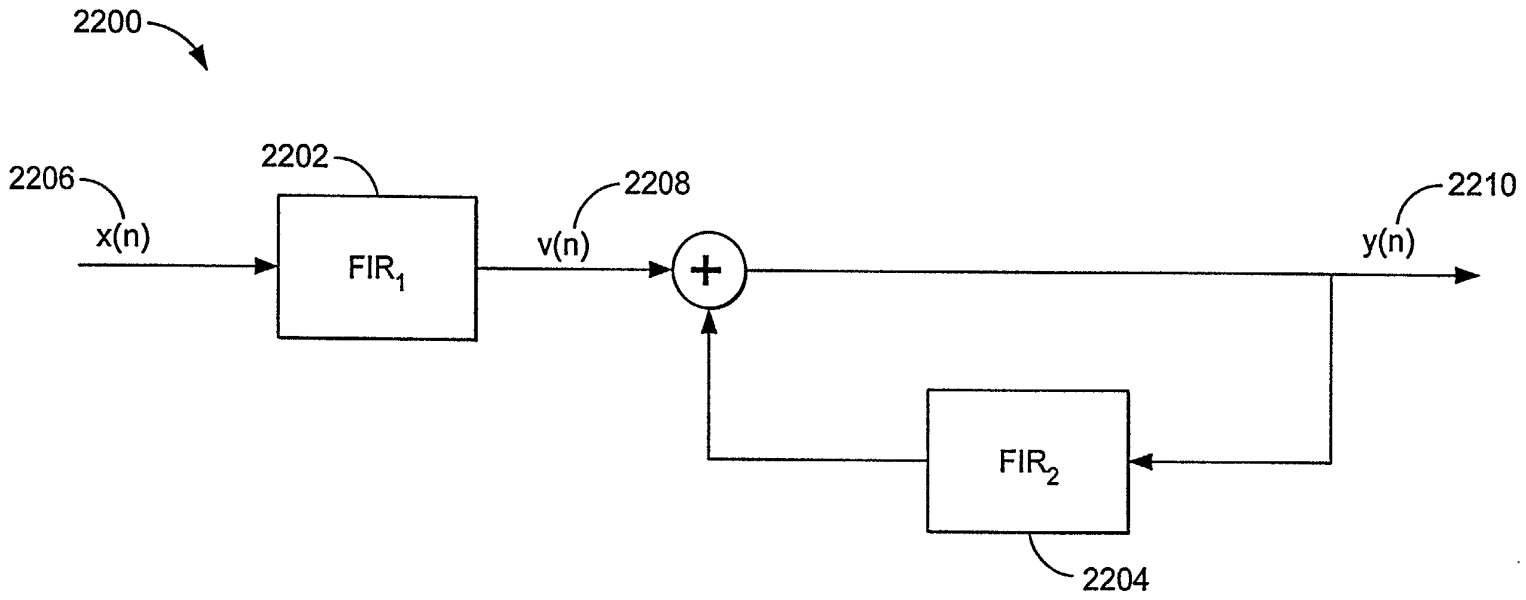


FIG. 22

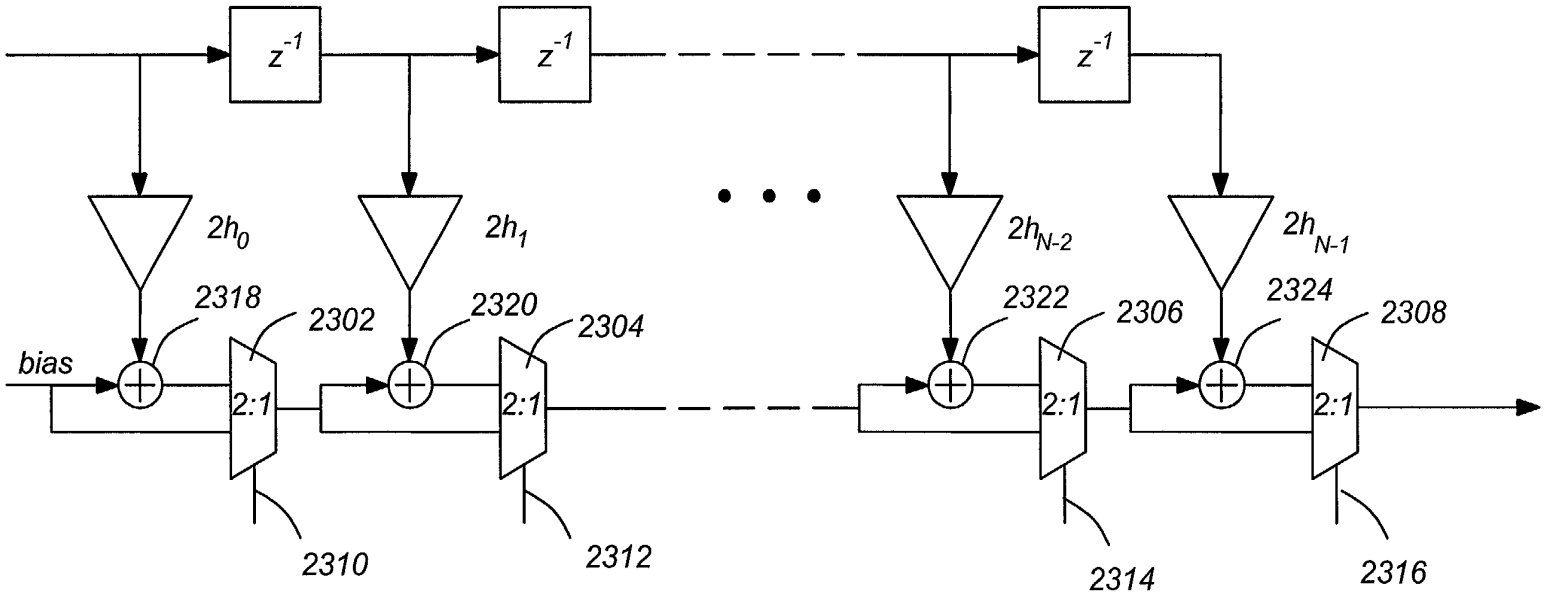

z^{-1}
z^{-1}
z^{-1}
$2h_0$
$2h_1$
$2h_{N-2}$
$2h_{N-1}$
2318
2302
2320
2304
2322
2306
2324
2308
bias
2:1
2:1
2:1
2:1
2310
2312
2314
2316

FIG. 23

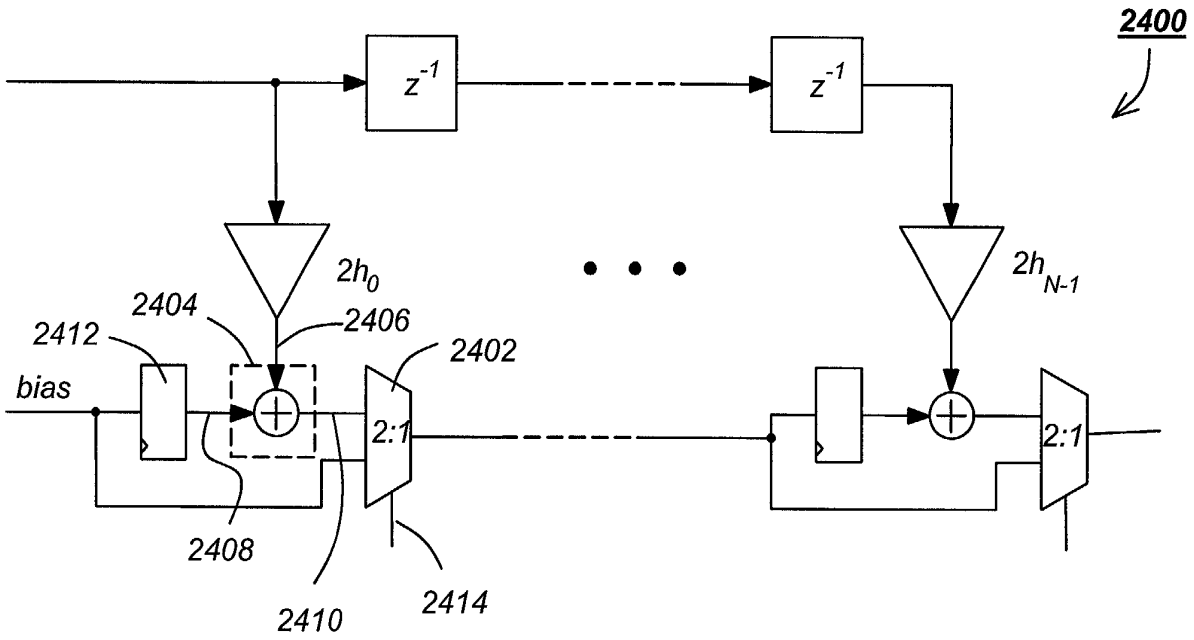

2400
z^{-1}
z^{-1}
$2h_0$
$2h_{N-1}$
2404
2406
2412
2402
bias
2:1
2:1
2408
2410
2414

FIG. 24

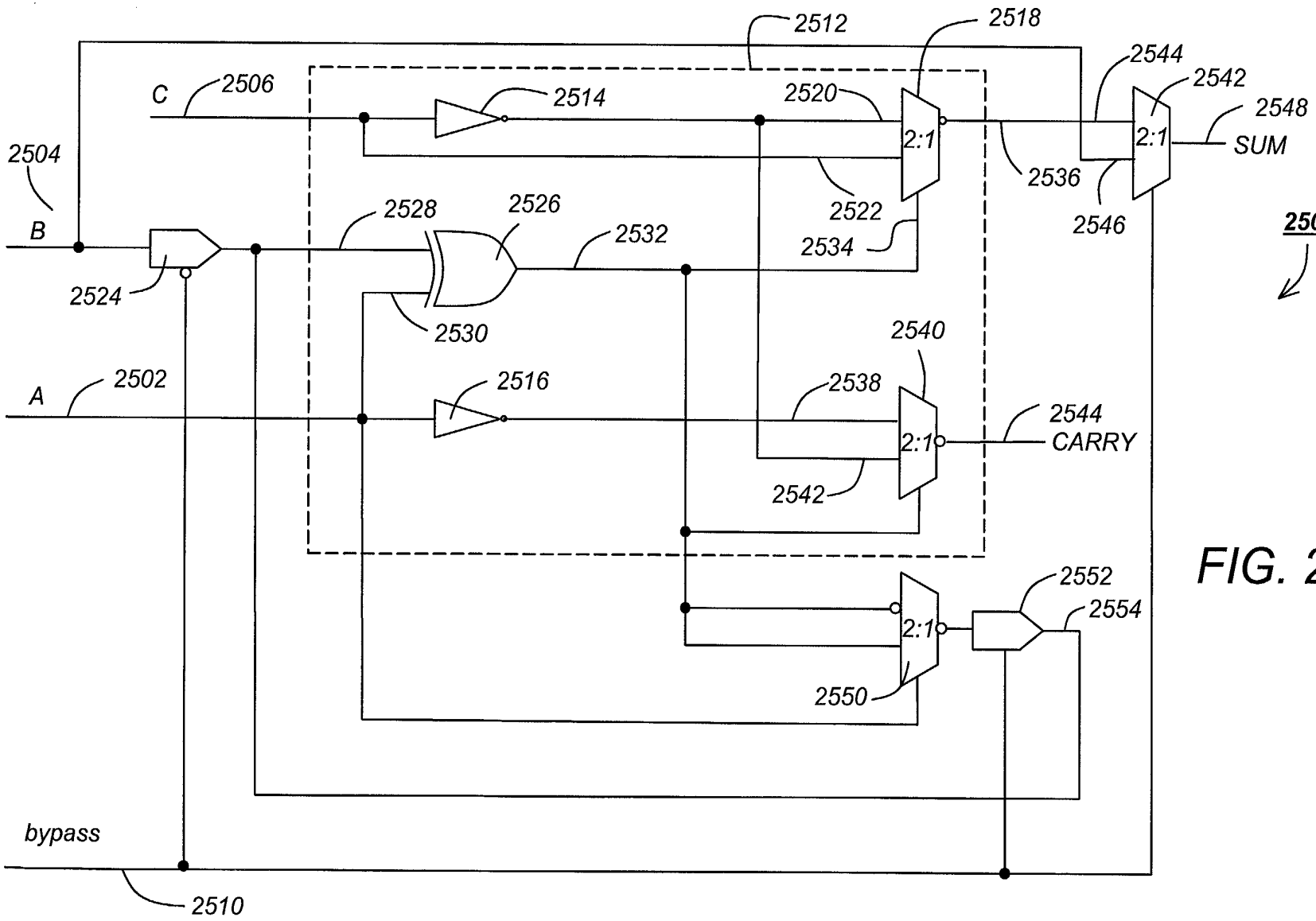

2512
2518
2544
2542
2548
SUM
C
2506
2514
2520
2504
2536
2522
2546
2500
B
2528
2526
2532
2534
2524
2530
2540
A
2502
2516
2538
2544
CARRY
2542
2552
2554
2550
bypass
2510

FIG. 25A

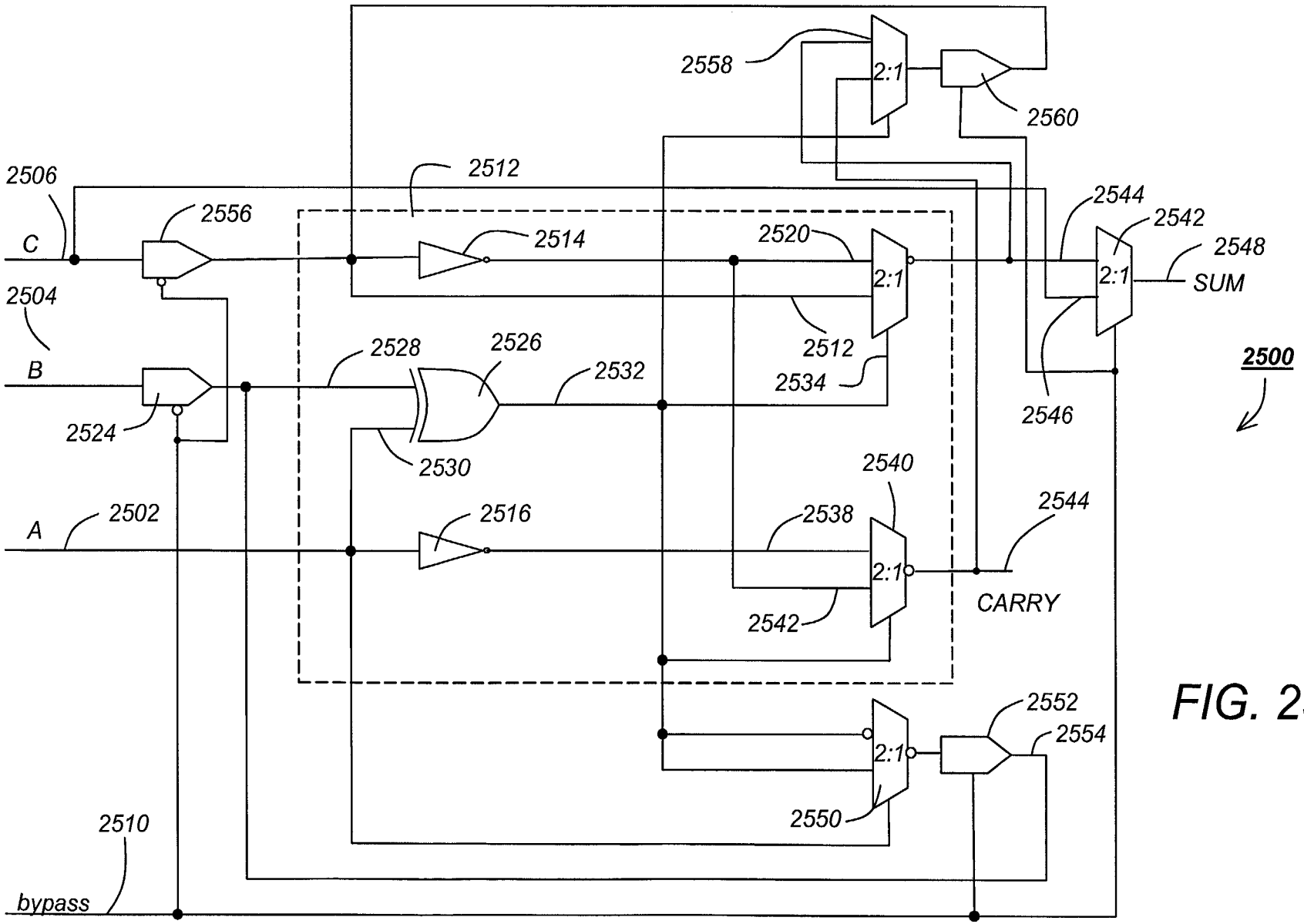

2558
2:1
2560
2506
2512
2544
2542
2556
C
2514
2520
2:1
2:1
2548
SUM
2504
2512
2534
2528
2526
2532
2500
B
2524
2530
2546
2540
2544
A
2502
2516
2538
2:1
CARRY
2542
2552
2554
2:1
2550
2510
bypass

FIG. 25B

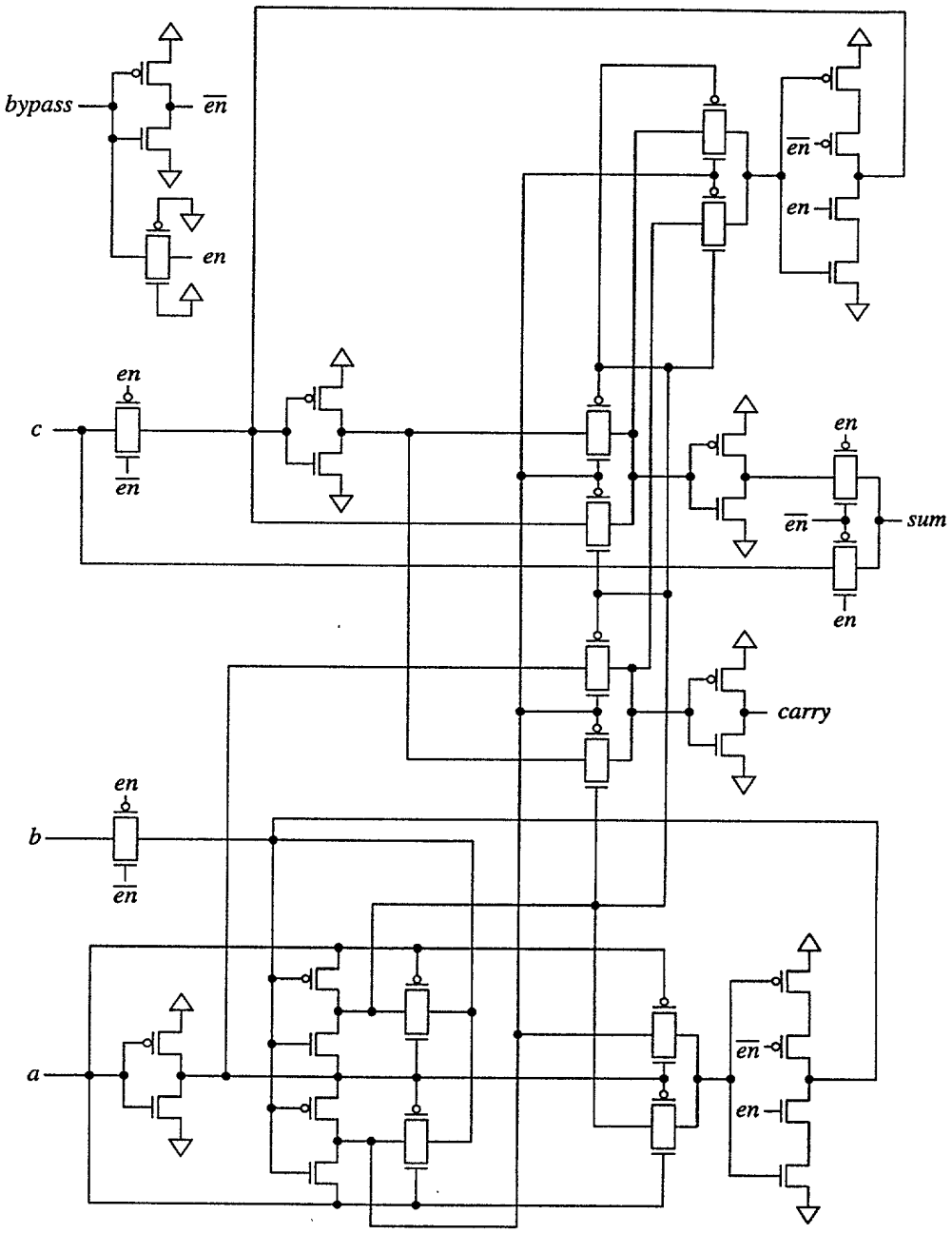


FIG. 25C

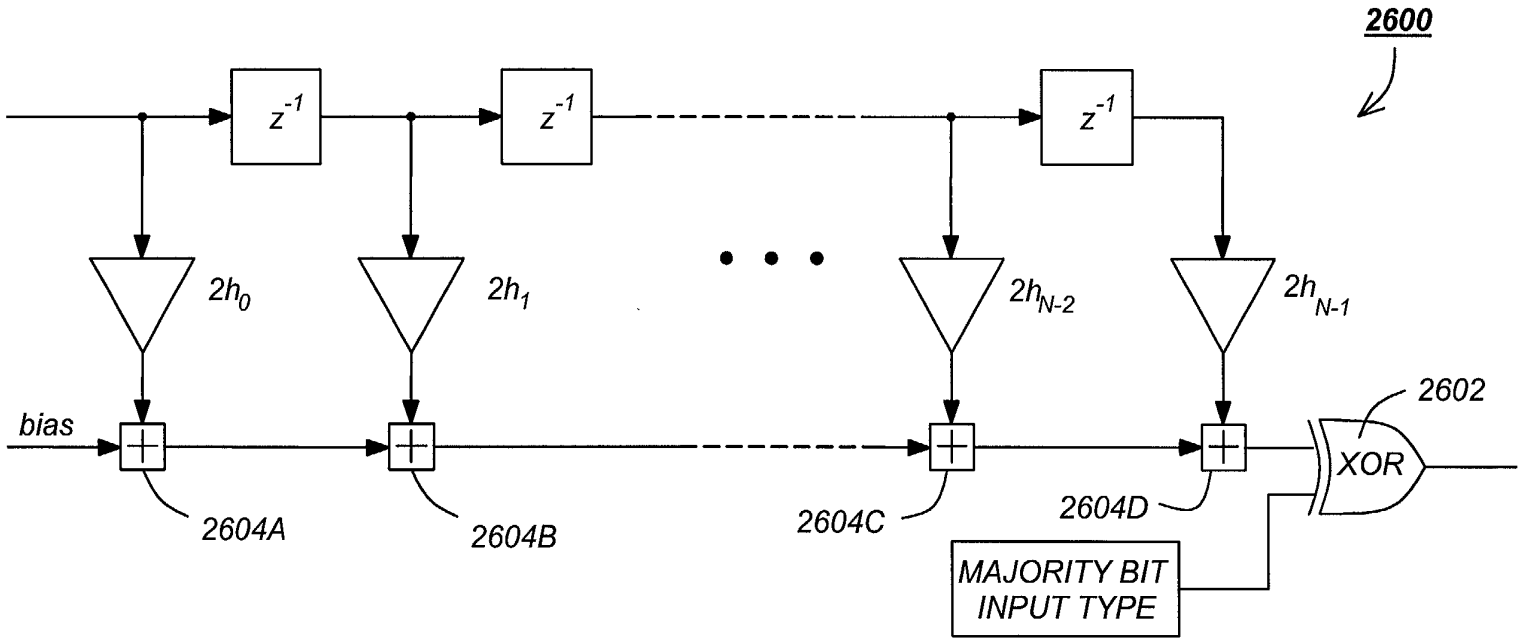

2600
z^{-1}
z^{-1}
z^{-1}
$2h_0$
$2h_1$
$2h_{N-2}$
$2h_{N-1}$
bias
2602
XOR
2604A
2604B
2604C
2604D
MAJORITY BIT INPUT TYPE

FIG. 26

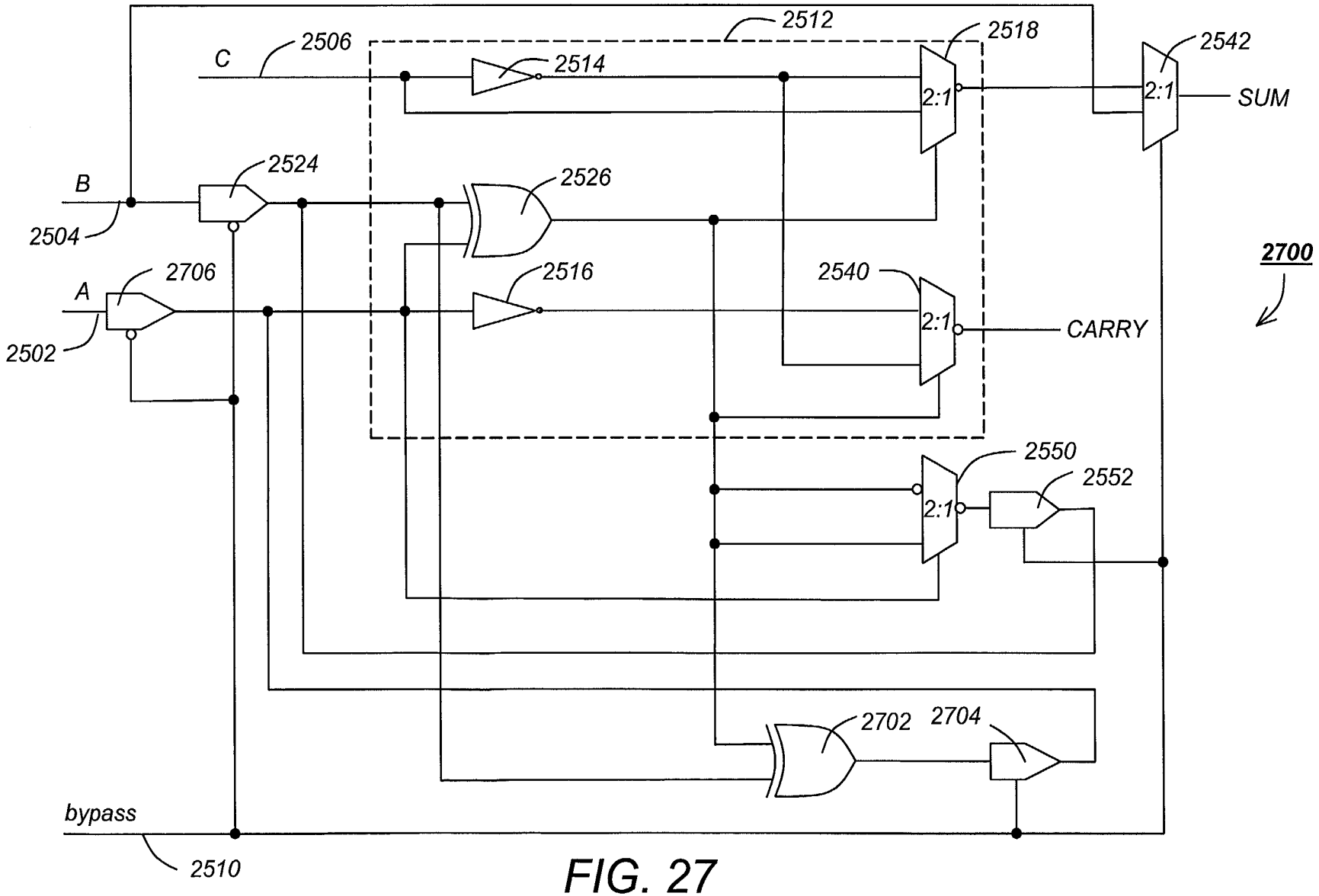

2512
2506
C
2514
2518
2542
2:1
2:1
SUM
2524
2526
B
2504
2700
2706
2516
2540
A
2:1
CARRY
2502
2550
2552
2:1
2702
2704
bypass
2510

FIG. 27

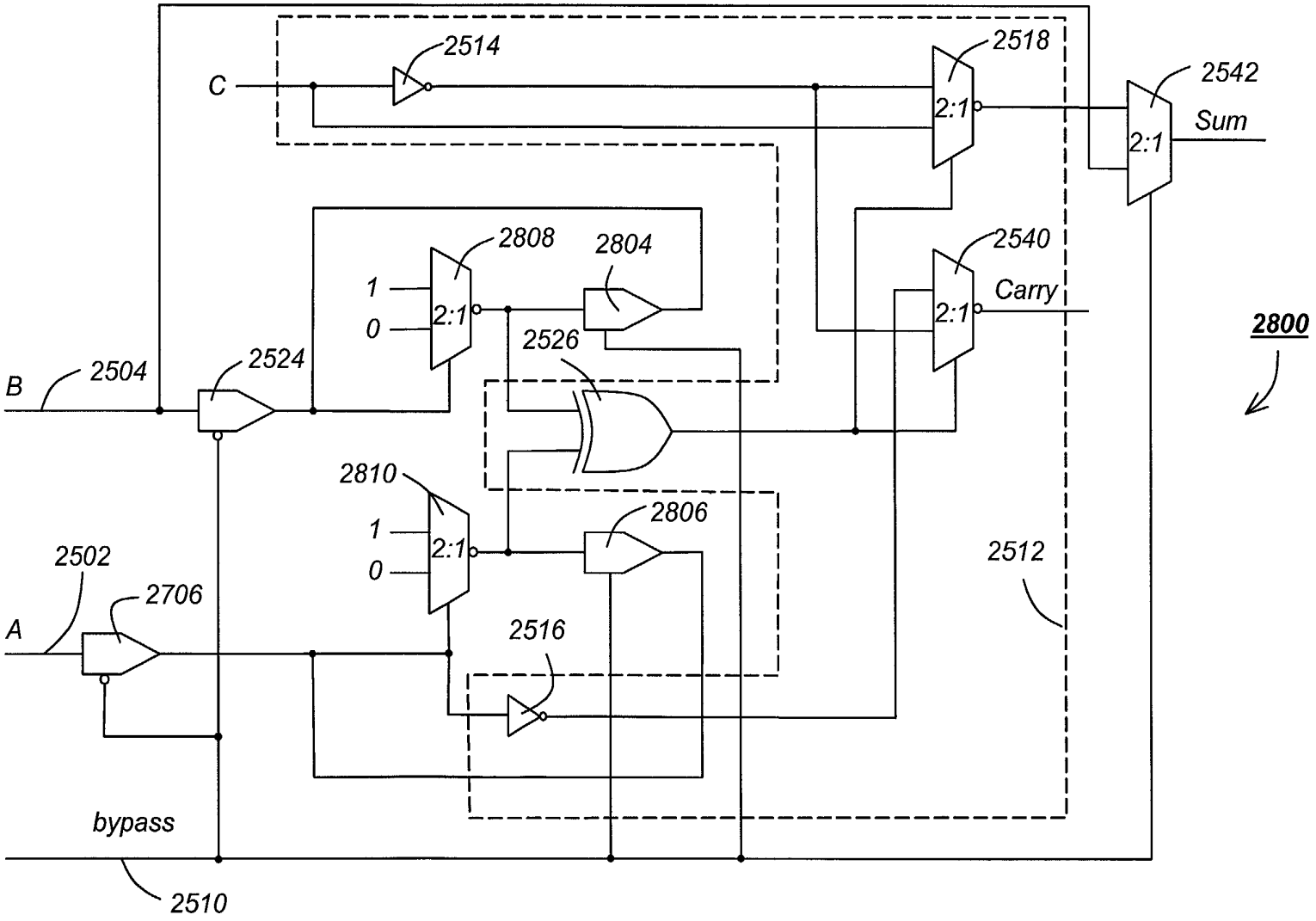

C
2514
2518
2542
2:1
Sum
2:1
2808
2804
2540
1
0
2:1
Carry
2:1
2800
B
2504
2524
2526
2810
1
0
2:1
2806
2502
2706
A
2516
2512
bypass
2510

FIG. 28

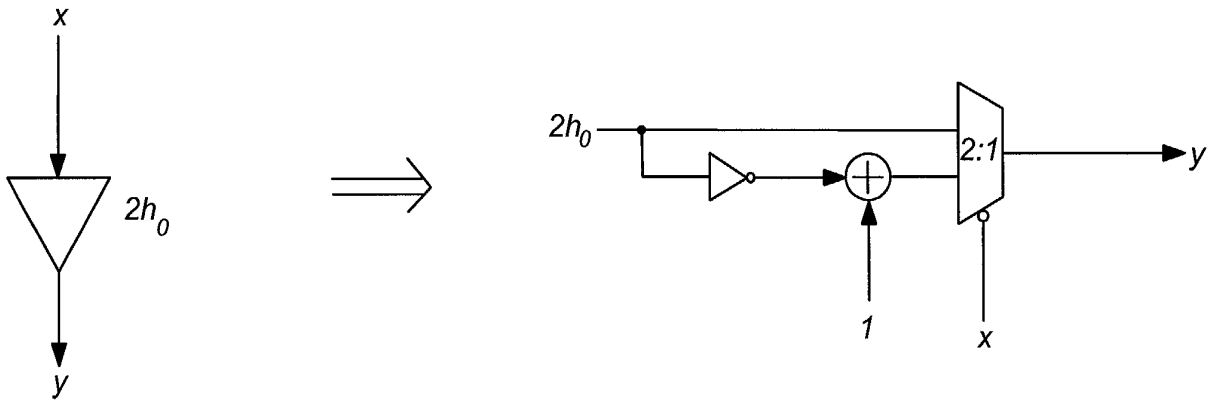
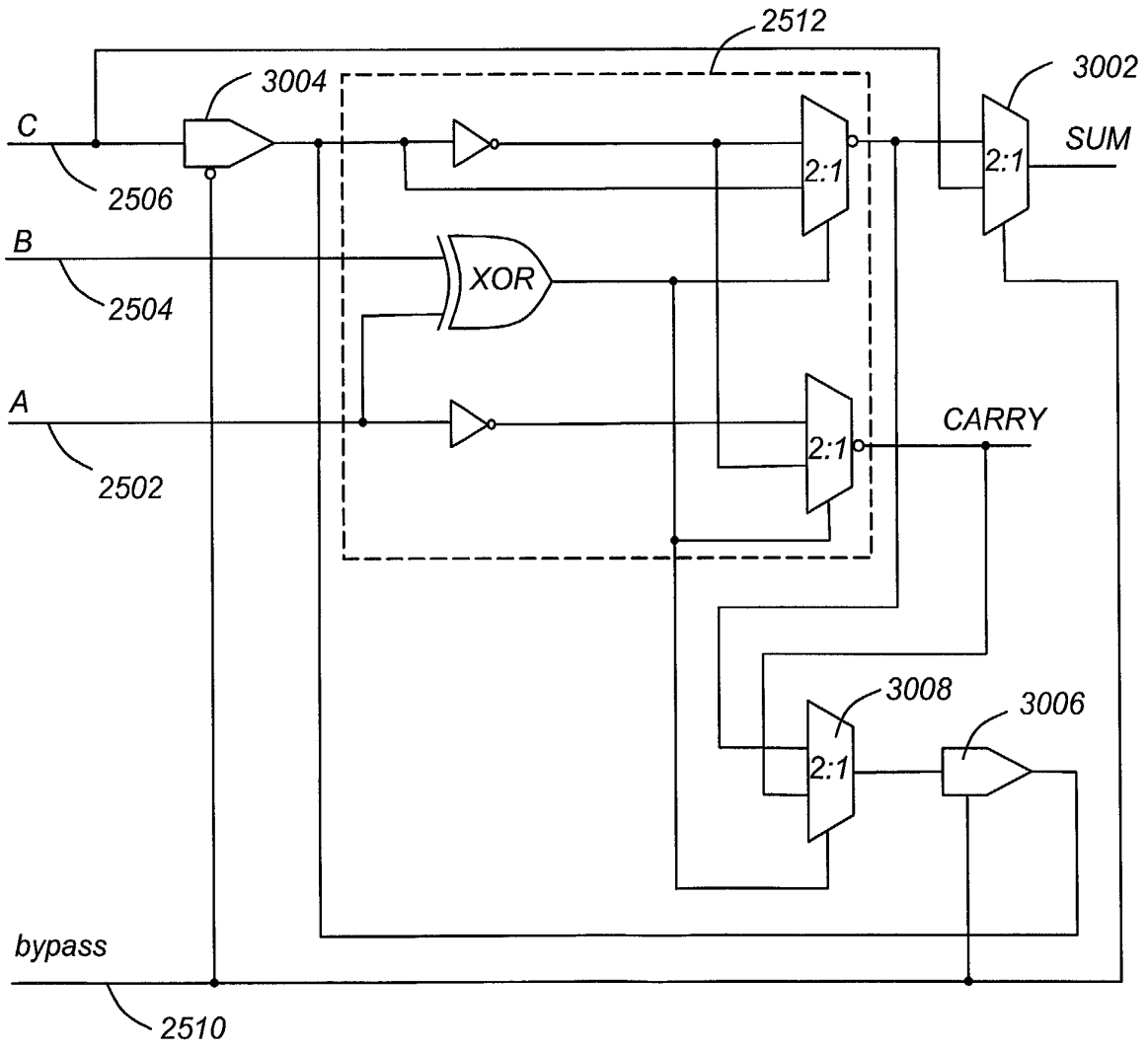


FIG. 29


2512
3004
3002
C
SUM
2506
2:1
2:1
B
XOR
2504
A
CARRY
2:1
2502
3008
3006
2:1
bypass
2510

3000

FIG. 30

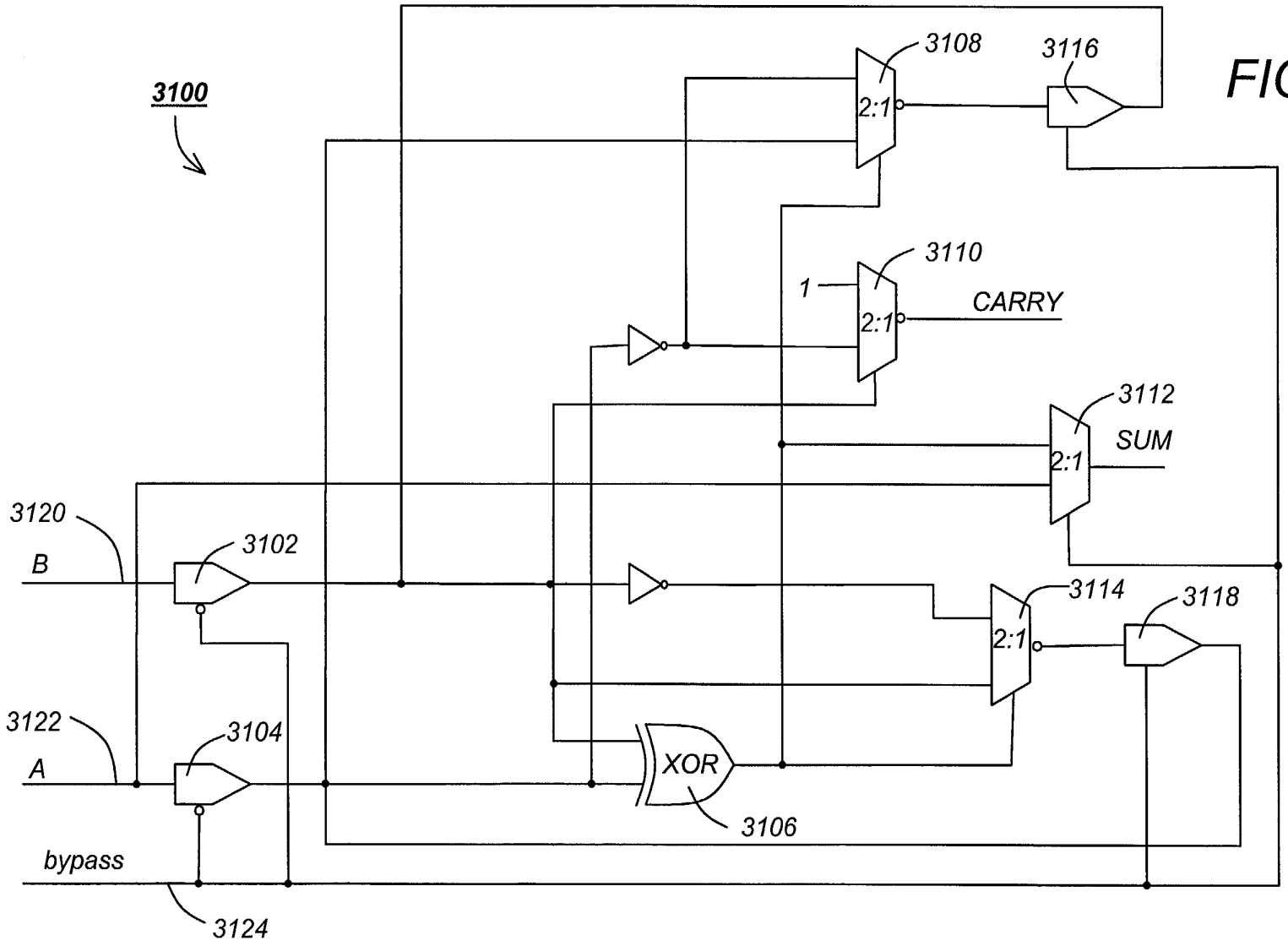

3100
3108
3116
2:1
3110
1
2:1
CARRY
3112
2:1
SUM
3120
B
3102
3114
3118
2:1
3122
A
3104
XOR
3106
bypass
3124

FIG. 31

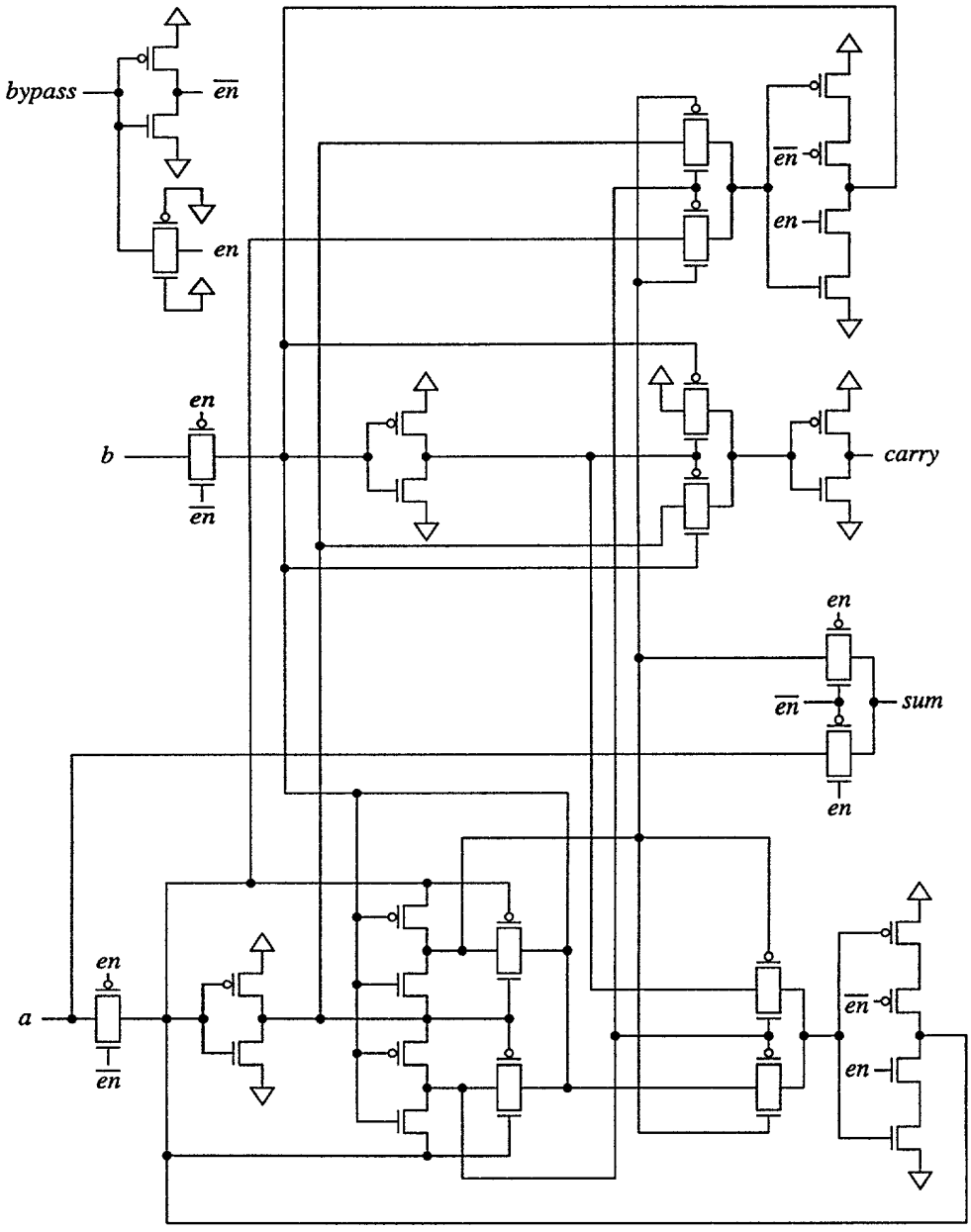

bypass
$\overline{en}$
en
b
en
$\overline{en}$
carry
en
$\overline{en}$
sum
en
a
en
$\overline{en}$
$\overline{en}$
en

FIG 32

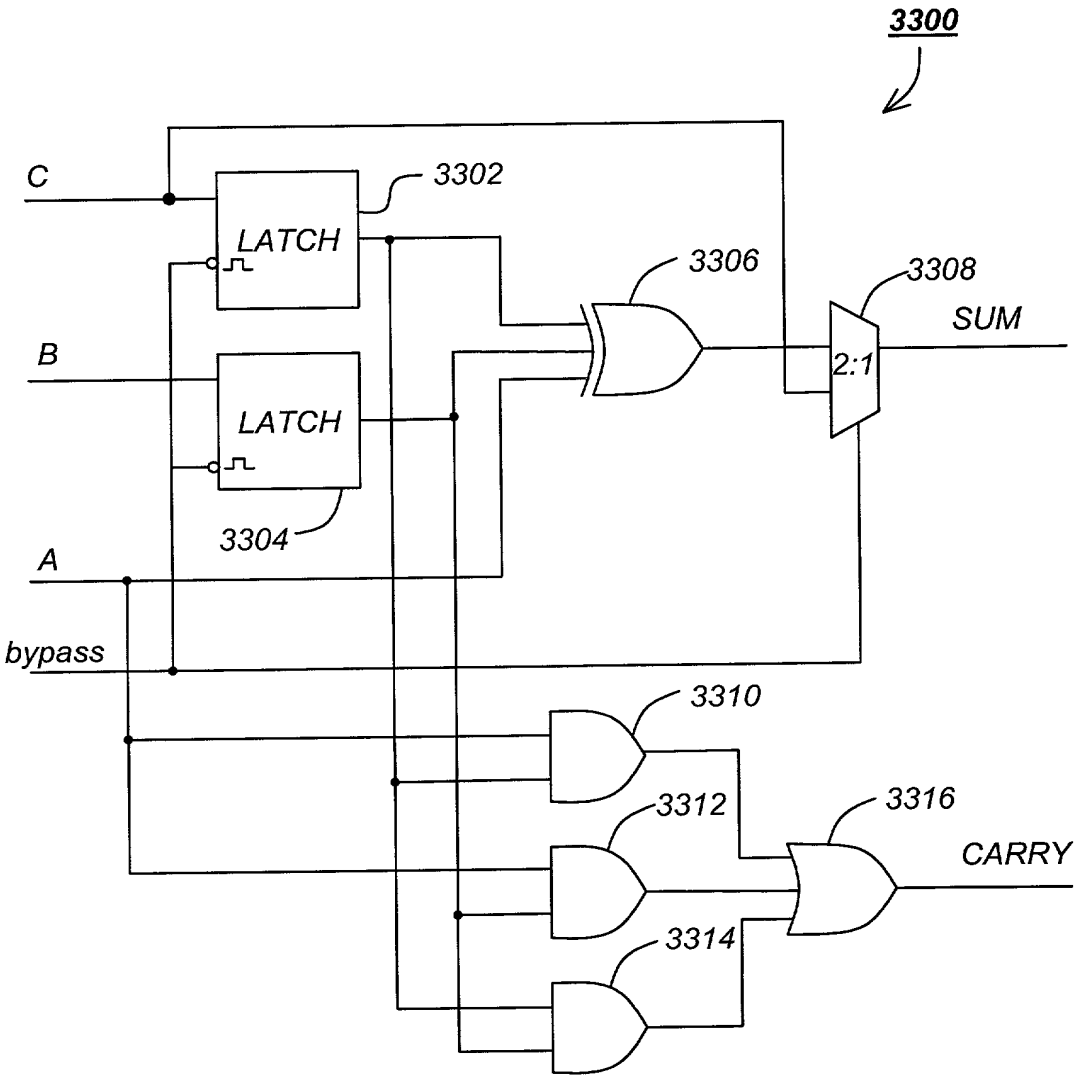

3300
C
LATCH
3302
B
LATCH
3304
A
bypass
3306
3308
SUM
2:1
3310
3312
3314
3316
CARRY

FIG. 33

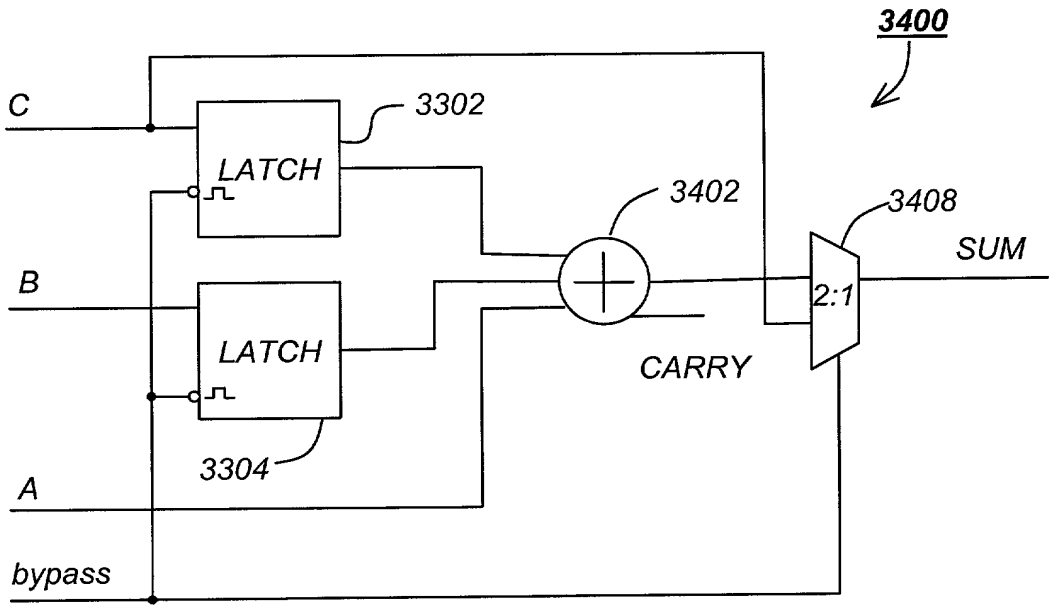


FIG. 34